

***An Efficient, Semi-Implicit
Pressure-Based Scheme
Employing a High-Resolution
Finite Element Method for
Simulating Transient and
Steady, Inviscid and Viscous,
Compressed Flows on
Unstructured Grids***

*Richard C. Martineau
Ray A. Berry*

April 2003

*Idaho National Engineering and Environmental Laboratory
Bechtel BWXT Idaho, LLC*



An Efficient, Semi-Implicit Pressure-Based Scheme Employing a High-Resolution Finite Element Method for Simulating Transient and Steady, Inviscid and Viscous, Compressible Flows on Unstructured Grids

Richard C. Martineau

and

Ray A. Berry

April 2003

This report documents original research performed by Richard C. Martineau and was accepted as a dissertation by the University of Idaho, Department of Mechanical Engineering, in partial fulfillment of the requirements for the degree of Doctor of Philosophy. This doctoral research effort was supervised by Ray A. Berry.

**Idaho National Engineering and Environmental Laboratory
Idaho Falls, Idaho 83415**

**Prepared for the
U.S. Department of Energy
Under DOE Idaho Operations Office
Contract DE-AC07-99ID13727**

ABSTRACT

A new semi-implicit pressure-based Computational Fluid Dynamics (CFD) scheme for simulating a wide range of transient and steady, inviscid and viscous compressible flow on unstructured finite elements is presented here. This new CFD scheme, termed the PCICE-FEM (Pressure-Corrected ICE-Finite Element Method) scheme, is composed of three computational phases, an explicit predictor, an elliptic pressure Poisson solution, and a semi-implicit pressure-correction of the flow variables. The PCICE-FEM scheme is capable of second-order temporal accuracy by incorporating a combination of a time-weighted form of the two-step Taylor-Galerkin Finite Element Method scheme as an explicit predictor for the balance of momentum equations and the finite element form of a time-weighted trapezoid rule method for the semi-implicit form of the governing hydrodynamic equations. Second-order spatial accuracy is accomplished by linear unstructured finite element discretization. The PCICE-FEM scheme employs Flux-Corrected Transport as a high-resolution filter for shock capturing. The scheme is capable of simulating flows from the nearly incompressible to the high supersonic flow regimes.

The PCICE-FEM scheme represents an advancement in mass-momentum coupled, pressure-based schemes. The governing hydrodynamic equations for this scheme are the conservative form of the balance of momentum equations (Navier-Stokes), mass conservation equation, and total energy equation. An operator splitting process is performed along explicit and implicit operators of the semi-implicit governing equations to render the PCICE-FEM scheme in the class of predictor-corrector schemes. The complete set of semi-implicit governing equations in the PCICE-FEM scheme are cast in this form, an explicit predictor phase and a semi-implicit pressure-correction phase with the elliptic pressure Poisson solution coupling the predictor-corrector phases. The result of this predictor-corrector formulation is that the pressure Poisson equation in the PCICE-FEM scheme is provided with sufficient internal energy information to avoid iteration. The ability of the PCICE-FEM scheme to accurately and efficiently simulate a wide variety of inviscid and viscous compressible flows is demonstrated here.

Table of Contents

1. Introduction	1
1.1 Governing Hydrodynamic Equations	4
1.2 Semi-Implicit Pressure-Based Schemes	8
1.2.1 Mass-Momentum Coupling	10
1.2.2 Energy-Momentum Coupling	13
1.2.3 Previous Semi-Implicit Pressure-Based Research Efforts	15
1.3 Specific Objectives of this Research	19
 2. The Explicit Two-Step Taylor-Galerkin Finite Element Method for Unstructured Grids (FEM-FCT)	 21
2.1 The High-Order Scheme: The Consistent-Mass Two-Step Taylor-Galerkin FEM	23
2.2 Stability of the Two-Step Taylor-Galerkin Finite Element Method	26
2.3 Convergence Acceleration for Steady-State Simulations	27
2.4 Stabilizing the High-Order Scheme with Artificial Dissipation	27
2.4.1 Lapidus Artificial Diffusion	29
2.4.2 Peraire's Artificial Dissipation	31
2.4.3 Flux-Corrected Transport	33
2.4.3.1 Flux-Corrected Transport for the Finite Element Method	33
2.4.3.2 The Low-Order Scheme: Lumped-Mass Taylor-Galerkin Plus Diffusion	34
2.4.3.3 Antidiffusive Element Contributions	35
2.4.3.4 The FCT Limiting Procedure	35
2.4.3.5 FCT Limiting for Systems of Equations	37
2.5 Boundary Conditions	37
2.5.1 Compatibility Relations for Compressible Flow	38
2.5.2 Spatial and Temporal Discretization	51
2.5.3 Free-Slip Solid Wall and Symmetry Boundary Conditions	61
2.5.4 Characteristic Boundary Conditions for Flow Exits	62
2.5.4.1 Subsonic Flow Exit with Specified Back Pressure	62
2.5.4.2 Non-Reflecting Flow Exit	63
2.5.5 Characteristic Boundary Conditions for Flow Inlets	65
2.5.5.1 Subsonic Inlet with Specified Boundary Pressure and Temperature	65
2.5.5.2 Subsonic Inlet with Specified Stagnation Pressure and Temperature	66
2.5.5.3 Mass Injection	67
2.5.6 Viscous Boundary Conditions	68
2.5.7 Simulation Examples with Strong Flow Interaction with the Boundary	68
2.5.7.1 Steady Mach 2.5 Channel with a 5 Degree Double Wedge Obstruction	69
2.5.7.2 Transient Mach Reflection of a Shock Wave from a 25 Degree Wedge	70
 3. The Pressure-Corrected ICE Finite Element Method for Unstructured Grids (PCICE-FEM)	 75
3.1 Temporal Discretization of the Governing Hydrodynamic Equations	80
3.2 The Explicit Predictor Phase	81
3.2.1 Explicit Predictor Components of the Semi-Implicit Equations	82
3.2.2 Finite Element Discretization of the Explicit Predictor Equations	84
3.2.2.1 Finite Element Discretization of the Explicit Predictor for the Balance of Momentum Equations	84
3.2.2.2 Finite Element Discretization of the Explicit Predictor for the Mass	

Conservation and Total Energy Equations	86
3.2.3 Boundary Conditions for the Explicit Predictor Equations	89
3.2.4 Solution of the Explicit Predictor Equations	92
3.2.5 Application of Artificial Dissipation for the Components of the Explicit Predictor Phase of the PCICE-FEM scheme	92
3.2.5.1 Lapidus Artificial Diffusion	93
3.2.5.2 Peraire's Artificial Dissipation	93
3.2.5.3 Flux-Corrected Transport	95
3.3 The Elliptic Pressure Poisson Equation	96
3.3.1 Derivation of the Pressure Poisson Equation	96
3.3.2 Finite Element Discretization of the Pressure Poisson Equation	100
3.3.3 Boundary Conditions for the Pressure Poisson Equation	103
3.3.4 Solving the Pressure Poisson Equation with Bi-CGSTAB	106
3.4 Pressure-Correction of the Semi-Implicit Conservation Equations	109
3.4.1 Derivation of the Pressure-Correction Equations	109
3.4.2 Finite Element Discretization and Solution of the Pressure-Correction Equations	111
3.4.3 The PCICE-FEM Algorithmic Steps	113
3.5 Stability of the PCICE-FEM Scheme	114
4. Results and Conclusions	115
4.1 Transient Simulations	117
4.1.1 The Shock Tube Problem	117
4.1.2 Laboratory Accident Simulation	119
4.1.3 Low-Speed Viscous Flow Around a Cylinder (Von Karman Vortex Street)	128
4.2 Steady-State Flow Simulations	131
4.2.1 Mach Reflection in a Channel from an 18 Degree Wedge Obstruction	131
4.2.2 Double-Throated Nozzle (The GAMM Problem)	140
4.2.3 Thermally Driven Flow in a Square Cavity	142
4.3 Conclusions	152
A: General Properties of Linear Triangular Finite Elements	165
A.1 Properties of Shape Functions for Linear Triangular Finite Elements	165
A.2 Evaluation of Basic Integrals Common to Linear Triangular Finite Elements	170
A.3 Boundary Face Outward Pointing Unit Normal Vector and Element Length	171
B: Determination of Boundary Node Outward Pointing Unit Normal Vector that is Consistent with the Principle of Mass Conservation	173

1. Introduction

The primary purpose of this research effort is to develop an efficient, high-resolution, high-order accurate, semi-implicit pressure-based scheme for simulating a wide range of transient and steady, inviscid and viscous compressible flows on unstructured grids. The research centers on the development of a semi-implicit scheme composed of three computational phases, an explicit predictor, an elliptic pressure Poisson solution, and a semi-implicit pressure-correction of the conserved flow variables. The pressure, momentum, and density variables in the governing hydrodynamic equations are treated in an implicit fashion. Hence, the formulation is referred to as semi-implicit. The three implicit variables are coupled by substituting the momentum equation into the mass conservation equation to eliminate momentum as an unknown. Density is then expressed in terms of pressure by employing the equation of state. This is known as mass-momentum coupling or pressure-velocity-density coupling. These substitutions result in a single second-order differential Poisson equation in terms of pressure (pressure Poisson). Thus, the scheme is commonly referred to as pressure-based. The solution of the pressure Poisson equation effectively solves the mass conservation and balance of momentum equations simultaneously. This semi-implicit treatment has two advantages over explicit schemes. The first advantage is that the acoustic component from the explicit time step size criteria is removed. The spatial stiffness of slow flows on small computational cells, typically found in viscous boundary layer discretizations, is then eliminated. The second advantage is that the pressure obtained with this semi-implicit treatment corrects the momentum to satisfy mass conservation requirements. Theoretically, this allows incompressible flows to be simulated with compressible flow equations. This type of scheme can then be used to simulate any flow from nearly incompressible to supersonic.

The typical mass-momentum coupled, semi-implicit, pressure-based scheme requires an iterative process to satisfy conservation of mass and energy requirements. The typical scheme does not solve the energy equation until a new pressure field is obtained from the pressure Poisson equation. However, for a compressible gas, pressure is a function of the density and the internal energy. The typical scheme first solves the pressure Poisson equation with information from an explicit (predictor) mass conservation and balance of momentum solution. A correction of mass and momentum is then determined with the new pressure field obtained from the pressure Poisson solution. The energy equation is then explicitly advanced in time with the new mass, momentum, and pressure. However, the new energy solution is generally not consistent with mass, momentum, and pressure because information from the new energy solution was required for full coupling of the energy equation with the pressure Poisson equation. Therefore, the updated (corrected) values of mass, momentum, and energy are then used to re-solve the pressure Poisson equation. A new pressure field is computed, from which, a new set of pressure corrected conserved variables are then determined. This iterative process continues until convergence on pressure is achieved.

The new semi-implicit pressure-based scheme presented here is derived so that the total energy equation is sufficiently coupled to the pressure Poisson equation in order to avoid iteration between the pressure Poisson equation and the implicit pressure-correction equations. Total energy is explicitly convected in a manner that is consistent with the explicit conservation of mass equation. The same explicit momentum components are used to convect

both mass and total energy. The pressure Poisson equation thus has the energy information it requires to yield an accurate implicit pressure. A pressure-corrected total energy equation is derived that is also consistent with the pressure-correction equation for mass conservation. At the end of a time step, the conserved values of mass, momentum, and total energy are all pressure-corrected. As a result, the iterative process is not required which greatly reduces computer simulation time. This aspect is highly advantageous when computing transient simulations that are highly compressible and/or contains high energy deposition, chemical reactions, or phase change. This gain in efficiency is also apparent when marching to steady-state, which results in faster convergence.

The explicit predictor phase of the new pressure-based scheme is adapted from an explicit scheme developed at the Naval Research Laboratory (NRL) and the University of Wales. This explicit scheme, called the Finite Element Method Flux-Corrected Transport (FEM-FCT), employs a two-step time integration based upon the second-order Taylor-Galerkin temporal discretization. The FEM-FCT scheme was developed primarily for computational fluid dynamics (CFD) problems on unstructured grids. Linear triangular elements (two-dimensional) and linear tetrahedral elements (three-dimensional) were employed for several reasons. The first is that they are easily generated on domains with complex geometries. Second, they provide second-order spatial accuracy. Third, they can be easily adapted to minimize error in the solution. And finally, they can be integrated exactly, which precludes the need of time consuming numerical integration. Flux-Corrected Transport (FCT) is a high-resolution method which limits the convective fluxes to avoid spurious oscillations in the solution near large changes in solution gradients (curvature). For example, in high-speed flows where shocks are present, oscillations in the solution variables are generated by the second-order space discretization. FCT suppresses these oscillations and maintains the positivity of the conserved variables, density and total energy. Low-speed flows can also contain extreme changes in flow variable gradients. The simulation of flames is a common problem where the convective velocities are low and the pressure is nearly constant, which indicates that the entire flow field is nearly incompressible. However, across the flame front, density can vary by several orders of magnitude. Second-order differencing will result in density oscillations across the flame front. FCT can be utilized to control these oscillations in density caused by the large changes in density gradient.

The mass-momentum coupling approach employed for the new pressure-based scheme results in a Poisson equation in terms of the change in pressure across a time step. This pressure Poisson equation is elliptic and requires a discretized algebraic system of equations to be simultaneously solved. The coefficient matrix derived from the spatial discretization of the governing partial differential equations is sparse. Also, the substitution of the equation of state for the partial derivative of density with respect to time has rendered the coefficient matrix to be non-symmetric. Direct methods, such as Gaussian elimination, are not applicable for solving large systems of equations. Direct methods require inversion and storage of the coefficient matrix and are extremely slow in actual computer time due to an excessively large amount of required computer operations. Furthermore, the spatial discretization error is generally much larger than the error obtained with direct methods, which can approach machine error (the absolute number of significant digits the machine is capable of). There is, therefore, no reason then to solve the system of equations with a direct method. Efficient

iterative methods that require several orders of magnitude less computation than direct methods have been developed for the type of problem presented here. The preconditioned Bi-Conjugate Gradient Stabilized (Bi-CGSTAB) iterative method was chosen for its ability to solve a nonsymmetric system of equations and for its smooth convergence. A new Jacobi preconditioner was developed for Bi-CGSTAB based upon the coefficient matrix for the pressure Poisson equation. With this new diagonal preconditioner and exact finite element integration, no matrices are required by Bi-CGSTAB, only the assembly of matrix-vector dot products. Also, this new preconditioner has proven to be very efficient for Bi-CGSTAB, requiring only 4-7 iterations to achieve a reduction of four orders of magnitude in the relative error for change in pressure.

The mass-momentum coupling technique introduced here is a descendent of the Implicit Continuous-fluid Eulerian (ICE) scheme originally developed at the Los Alamos National Laboratory (LANL) in the late 1960s. In fact, it is the basis for the successful development of the new semi-implicit pressure-based scheme that, from this point on, will be referred to as the PCICE-FEM (Pressure-Corrected ICE-Finite Element Method) scheme. The PCICE-FEM scheme incorporates a combination of a time-weighted form of the two-step Taylor-Galerkin scheme for the explicit balance of momentum equations and a time-weighted trapezoid rule method for the semi-implicit equations to achieve second-order temporal differencing. Second-order spatial differencing is accomplished by linear unstructured finite element discretization. For documentation purposes, this study illustrates the PCICE-FEM development on two-dimensional flow fields. Therefore, linear triangular finite elements will be used exclusively for the finite element formulations. The technique presented here employing mass-momentum coupling with full inclusion of energy effects for the PCICE-FEM scheme could be used with other spatial differencing and high-resolution schemes. The FEM-FCT scheme was chosen as the basis for the explicit predictor phase of PCICE-FEM because of its known ability to accurately simulate strong transients (shock wave propagation) on complex geometries. The PCICE-FEM extends this capability. It excels on steady-state simulations and is applicable over a much wider Mach number range than the FEM-FCT scheme while maintaining the ability to accurately simulate strong transients.

Background information is required before discussing the PCICE-FEM scheme in detail. The rest of Chapter 1 and all of Chapter 2 will be devoted to providing this information. Of prime importance is how a semi-implicit pressure-based scheme for compressible flow is developed. First, the governing hydrodynamic equations for this research will be defined. Then, simple derivations of the two main coupling techniques for pressure-based schemes, mass-momentum coupling and energy-momentum coupling, will be presented. Following these derivations, an overview of previous research efforts in pressure-based schemes will be covered. This will allow for discussion of how the PCICE-FEM differs from and improves upon previous efforts. The end of Chapter 1 will state the specific objectives of this research and the resulting contributions to the field of computational fluid dynamics. Chapter 2 will provide an in-depth discussion of the explicit two-step Taylor-Galerkin finite element method with FCT, the FEM-FCT scheme. This will be necessary to provide the basis for the explicit predictor phase of the PCICE-FEM scheme. Also, a new characteristic method for applying boundary conditions on unstructured grids for the FEM-FCT scheme will be presented.

Chapters 3 and 4 will be devoted to the development and application of the PCICE-FEM scheme. In Chapter 3, a complete derivation of the PCICE-FEM scheme will be presented. This will include temporal discretization of the governing hydrodynamic equations and the three computational phases of the PCICE-FEM scheme, the components of the explicit predictor, the elliptic change in pressure Poisson equation, and the implicit pressure-correction of the conserved flow variables. A detailed development of the boundary conditions for these three phases will also be provided for closure. Chapter 4 will present the results and conclusions of this research. Simulations will be illustrated that span the entire range of applicability of the PCICE-FEM scheme. Specifically, this range will vary from the near incompressible to the supersonic flow regimes. This will include steady and transient, inviscid and viscous simulations. A basis for accuracy will be established by simulating flow for which there are known solutions. An Appendix containing the definitions of the finite element formulations and discretizations and a Bibliography containing the references incorporated into this research will conclude the documentation of this study.

1.1 Governing Hydrodynamic Equations

The governing hydrodynamic equations under consideration for this research are the compressible Euler (inviscid) and Navier-Stokes (viscous) equations. The governing equations can be defined in two-dimensional, differential form as

$$\frac{\partial U}{\partial t} + \frac{\partial f}{\partial x} + \frac{\partial g}{\partial y} = \frac{\partial f_v}{\partial x} + \frac{\partial g_v}{\partial y} + Q \quad (1.1)$$

or in compact differential vector form,

$$\frac{\partial U}{\partial t} + \vec{\nabla} \cdot \vec{F} = \vec{\nabla} \cdot \vec{F}_v + Q. \quad (1.2)$$

In equation (1.2), U is the column vector of the conservative variables given by

$$U = \begin{Bmatrix} \rho \\ \rho \vec{u} \\ \rho e_t \end{Bmatrix} = \begin{Bmatrix} \rho \\ \rho u \\ \rho v \\ \rho e_t \end{Bmatrix},$$

where ρ is density, u and v are the x and y components of the velocity vector \vec{u} , and e_t is the specific total energy. In terms of conserved variables, ρu and ρv are the momentum components and ρe_t is the total energy. The advective flux vector, \vec{F} in equation (1.2), is defined by its components f and g as

$$f = \begin{Bmatrix} \rho u \\ \rho u^2 + P \\ \rho u v \\ \rho u H \end{Bmatrix} \quad g = \begin{Bmatrix} \rho v \\ \rho u v \\ \rho v^2 + P \\ \rho v H \end{Bmatrix},$$

where H is the specific total enthalpy and P is the pressure. The viscous flux vector, \vec{F}_v in equation (1.2), is defined by its components f_v and g_v as

$$f_v = \begin{Bmatrix} 0 \\ \tau_{xx} \\ \tau_{xy} \\ u\tau_{xx} + v\tau_{xy} - q_x \end{Bmatrix} \quad g_v = \begin{Bmatrix} 0 \\ \tau_{xy} \\ \tau_{yy} \\ u\tau_{xy} + v\tau_{yy} - q_y \end{Bmatrix},$$

where τ_{xx} , τ_{xy} , and τ_{yy} are the components of the viscous stress tensor and q_x and q_y are the components of the heat flux vector. For a Newtonian fluid with Stokes hypothesis, the viscous stress tensor components are defined as

$$\tau_{xx} = \frac{2}{3}\mu \left(2\frac{\partial u}{\partial x} - \frac{\partial v}{\partial y} \right),$$

$$\tau_{xy} = \mu \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right),$$

and

$$\tau_{yy} = \frac{2}{3}\mu \left(2\frac{\partial v}{\partial y} - \frac{\partial u}{\partial x} \right),$$

where μ is the dynamic viscosity coefficient. Following Fourier's law of heat conduction, the vector components of the heat flux vector are

$$q_x = -k \frac{\partial T}{\partial x}$$

$$q_y = -k \frac{\partial T}{\partial y},$$

where k is the coefficient of thermal conductivity and T is the absolute temperature. Q in equation (1.2) is the column vector of source terms. Many terms can be included in Q , such as a mass source, momentum drag terms, and internal energy generation.

The mathematical classification of the governing equations of compressible flow is connected to the concept of characteristics, which are defined as families of surfaces in unsteady flow, along which certain combinations of variables remain constant or certain derivatives can

become discontinuous. A system of first-order partial differential equations are defined as hyperbolic if its homogenous part allows wave-like solutions. The left-hand side of equation (1.2), commonly referred to as the Euler equations, are strictly hyperbolic in space and time. The behavior of the mathematical properties of this system of equations are dominated by wave-like solutions. This is of prime importance when addressing boundary conditions (see Chapter 2) where the wave-like behavior dictates what specified information is necessary at the boundary in order to satisfy the governing equations. When the right-hand side of equation (1.2) is included, the wave-like behavior may be damped and the equation system is parabolic in nature. If all waves are damped, the system of equations is classified as elliptic. Therefore, the system of time-dependent Navier-Stokes equations is parabolic in time and space, although the conservation of mass equation is hyperbolic in nature. The time-dependent Navier-Stokes equations are then classified as mixed parabolic-hyperbolic. Incorporating the same reasoning, the steady-state form of the Navier-Stokes equations is then classified as mixed elliptic-hyperbolic.

Because of its wide range of applicability, the ideal gas equation of state will be used throughout this research for the derivation of characteristic boundary conditions and the development of the PCICE-FEM scheme. In most instances, a compressible gas can be considered as an ideal gas, even if viscous (real gas) effects are taken into account. The ideal gas equation of state is

$$P = \rho R_c T, \quad (1.3)$$

where R_c is the gas constant per unit mass and is equal to the universal gas constant divided by the molecular mass of the fluid. R_c is related to the specific heat coefficient at constant pressure by

$$c_p = \frac{\gamma}{\gamma - 1} R_c,$$

where γ is the ratio of specific heats,

$$\gamma = \frac{c_p}{c_v},$$

and c_v is the specific heat at constant volume. The internal energy e and the enthalpy h are only functions of temperature and have the following relations

$$e = c_v T = \frac{1}{\gamma - 1} \frac{P}{\rho} \quad (1.4)$$

and

$$h = c_p T = \frac{\gamma}{\gamma - 1} \frac{P}{\rho}. \quad (1.5)$$

The stagnation variables can be derived from the total enthalpy H , which can be defined as

$$H = \frac{\rho e_t + P}{\rho} = h + \frac{\vec{u}^2}{2} = c_p T_o, \quad (1.6)$$

where the total or stagnation temperature T_o is defined by

$$T_o = T + \frac{\vec{u}^2}{2c_p} = T \left(1 + \frac{\gamma - 1}{2} M^2 \right). \quad (1.7)$$

In equations (1.7), the Mach number M is the ratio of the magnitude of velocity to the sound speed,

$$M = \frac{|\vec{u}|}{c}.$$

The sound speed of the fluid is the change in pressure with respect to density at constant entropy. For an ideal gas, this translates into

$$c^2 = \left(\frac{\partial P}{\partial \rho} \right)_s = \gamma R_c T = \frac{\gamma P}{\rho}, \quad (1.8)$$

where the subscript s indicates constant entropy. The stagnation pressure can then be represented in terms of the stagnation temperature and Mach number as

$$P_o = P \left(\frac{T_o}{T} \right)^{\frac{\gamma}{\gamma - 1}} = P \left(1 + \frac{\gamma - 1}{2} M^2 \right)^{\frac{\gamma}{\gamma - 1}}. \quad (1.9)$$

The specific total energy can now be written in terms of the stagnation temperature or in terms of the sum of internal energy and kinetic energy,

$$e_t = c_v T_o = e + \frac{\vec{u}^2}{2}. \quad (1.10)$$

The static pressure and temperature can now be expressed in terms more useful for CFD applications. In terms of the solution variables, the pressure and temperature are

$$P = (\gamma - 1) \left(\rho e_t - \frac{\rho \vec{u} \cdot \vec{u}}{2} \right) \quad (1.11)$$

and

$$T = \frac{1}{c_v} \left(e_t - \frac{\vec{u} \cdot \vec{u}}{2} \right), \quad (1.12)$$

respectively.

1.2 Semi-Implicit Pressure-Based Schemes

Pressure-based schemes have been successfully developed for a wide variety of flows and discretizations. These flows range from isothermal incompressible to highly compressible flows, such as hypersonic chemically reacting flows. Depending on the type of flow and the application, pressure-based schemes have been formulated for implicit and semi-implicit temporal discretizations. This study will focus mainly upon semi-implicit formulations for compressible flow. A semi-implicit pressure-based scheme for compressible flow requires that pressure, momentum, and density (or total energy) be treated implicitly in order for the acoustic component to be removed from the explicit stability requirement. This section will attempt to give some basic understanding of pressure-based schemes by deriving pressure-based formulations incorporating both mass-momentum coupling and energy-momentum coupling for a simplified compressible flow equation set. An overview of previously developed pressure-based schemes will then be covered.

For illustrative purposes, the two-dimensional Euler equations will be employed to derive the basic semi-implicit pressure-based schemes. Neglecting the right-hand side of equation (1.2) and expanding the components yields;

for conservation of mass,

$$\frac{\partial \rho}{\partial t} = -\frac{\partial \rho u}{\partial x} - \frac{\partial \rho v}{\partial y}, \quad (1.13)$$

for the balance of momentum in the x-direction,

$$\frac{\partial \rho u}{\partial t} = -\frac{\partial \rho u^2}{\partial x} - \frac{\partial \rho uv}{\partial y} - \frac{\partial P}{\partial x}, \quad (1.14)$$

for the balance of momentum in the y-direction,

$$\frac{\partial \rho v}{\partial t} = -\frac{\partial \rho uv}{\partial x} - \frac{\partial \rho v^2}{\partial y} - \frac{\partial P}{\partial y}, \quad (1.15)$$

and for conservation of total energy,

$$\frac{\partial \rho e_t}{\partial t} = -\frac{\partial}{\partial x} \left[\rho u \left(\frac{\rho e_t + P}{\rho} \right) \right] - \frac{\partial}{\partial y} \left[\rho v \left(\frac{\rho e_t + P}{\rho} \right) \right]. \quad (1.16)$$

For the basic pressure-based formulations, the partial derivative of the conserved variables with respect to time are forward-time discretized and the semi-implicit pressure-based requirement that pressure, momentum, and density (or total energy) be implicit in time are applied to equations (1.13)-(1.16),

$$\frac{\rho^{n+1} - \rho^n}{\Delta t} = -\frac{\partial \rho u^{n+1}}{\partial x} - \frac{\partial \rho v^{n+1}}{\partial y}, \quad (1.17)$$

$$\frac{\rho u^{n+1} - \rho u^n}{\Delta t} = -\frac{\partial \rho u^2}{\partial x} \Big|_n - \frac{\partial \rho uv}{\partial y} \Big|_n - \frac{\partial P^{n+1}}{\partial x}, \quad (1.18)$$

$$\frac{\rho v^{n+1} - \rho v^n}{\Delta t} = -\frac{\partial \rho uv}{\partial x} \Big|_n - \frac{\partial \rho v^2}{\partial y} \Big|_n - \frac{\partial P^{n+1}}{\partial y}, \quad (1.19)$$

and

$$\frac{\rho e_t^{n+1} - \rho e_t^n}{\Delta t} = -\frac{\partial}{\partial x} \left[\rho u^{n+1} \left(\frac{\rho e_t^n + P^{n+1}}{\rho^{n+1}} \right) \right] - \frac{\partial}{\partial y} \left[\rho v^{n+1} \left(\frac{\rho e_t^n + P^{n+1}}{\rho^{n+1}} \right) \right]. \quad (1.20)$$

In equations (1.17)-(1.20), the superscripts n and $n+1$ denote values at the beginning and ending of a time step, respectively, and Δt is the incremental time step size. The superscripts on the conserved variables, ρu , ρv , and ρe_t , denote the time level for the whole variable, not just u , v , and e_t .

Common to both mass-momentum coupling and energy-momentum coupling is a pressure-correction equation for the balance of momentum. It is referred to as the momentum pressure-correction equation because, once a new implicit pressure field P^{n+1} has been obtained, the implicit momentum field is “corrected” to be consistent with the new pressure field. The

momentum pressure-correction equation can be expressed by casting equations (1.18) and (1.19) into

$$\rho \vec{u}^{n+1} - \rho \vec{u}^n = \vec{S} - \Delta t \vec{\nabla} P^{n+1}, \quad (1.21)$$

where \vec{S} is defined as

$$\vec{S} = -\Delta t \vec{\nabla} \cdot (\rho \vec{u} \otimes \vec{u})^n. \quad (1.22)$$

For this first-order temporal discretization, \vec{S} is composed only of information available at the beginning of a time step. It is essentially an explicit momentum flux equation that must be corrected with pressure to give the new time momentum components. As will be seen in Chapter 3, the PCICE-FEM scheme will be formulated with an \vec{S} that also contains time-advanced explicit terms arising from the second-order temporal discretization, momentum source terms, and the momentum components of the viscous stress tensor. In fact, \vec{S} is used as the explicit predictor for momentum in the PCICE-FEM scheme as well as in the original ICE scheme (Harlow and Amsden, 1971). It is an explicit momentum equation without the gradient of pressure. Employed as an explicit predictor, \vec{S} is often referred to as a fractional step for the balance of momentum (Chorin, 1967). In an iterative pressure-based scheme, \vec{S} is held constant in time for a given time step.

As with the momentum pressure-correction equation, the following coupled forms are independent of a chosen spatial discretization. However, a rigorous application of boundary conditions must be considered when formulating a pressure-based scheme. This was found to be especially true when developing the PCICE-FEM scheme. With an integral formulation of the spatially discretized equations, consistency must be maintained between the boundary conditions applied to pressure and momentum.

1.2.1 Mass-Momentum Coupling

One of the main ideas behind mass-momentum coupling is to obtain pressure and momentum fields that satisfies conservation of mass requirements. This is critical for an incompressible flow field where the conservation of mass equation is never actually solved because the density is defined to be constant along a fluid particle path. Instead, the divergence of velocity is incorporated as an algebraic constraint that is required to vanish in the incompressible limit. Conservation of mass requirements are no less important for compressible flow fields. Not only must the momentum field be consistent with the pressure field, but the momentum must also satisfy the conservation of mass equation. This then couples the variation in density in a compressible flow to the pressure field. In order to satisfy conservation of mass requirements, another relation between density and pressure is required. This relation is provided by the equation of state. This is the concept behind pressure-velocity-density (and thus momentum) coupling.

Mass-momentum coupling is achieved through the resulting pressure Poisson equation. This does not imply that the total energy equation is not coupled to the mass-momentum system. In fact, total energy is coupled to the mass conservation equation and the momentum balance equation through the equation of state. The equation of state is a thermodynamic relation connecting the density, internal energy, and pressure together. Total energy is the sum of internal energy and kinetic energy in an ideal compressible gas. The basic mass-momentum pressure-based scheme derived below will demonstrate how the governing hydrodynamic equations are indeed coupled through the equation of state.

The first stage in deriving a mass-momentum coupled pressure-based scheme is to substitute the balance of momentum equations into the mass flux components of the mass conservation equation, hence mass-momentum coupled. This substitution eliminates momentum as an unknown variable. Substituting equation (1.21) into equation (1.17) yields,

$$\rho^{n+1} - \rho^n = -\Delta t \vec{\nabla} \cdot (\rho \vec{u}^n + \vec{S}) + \Delta t^2 \vec{\nabla} \cdot \vec{\nabla} P^{n+1}. \quad (1.23)$$

Equation (1.23) is a Poisson equation in terms of the implicit pressure with the explicit mass flux term and the increment in density serving as a source term. This equation contains two unknowns, ρ^{n+1} and P^{n+1} . As discussed above, the equation of state supplies the relationship required to express density in terms of pressure. From the equation of state, equation (1.3), an incremental change in pressure can be defined as

$$\delta P = \left(\frac{\partial P}{\partial \rho} \right)_T \delta \rho + \left(\frac{\partial P}{\partial T} \right)_\rho \delta T. \quad (1.24)$$

Expanding the incremental change in pressure for an ideal gas about the values at the beginning of the time step gives

$$P^{n+1} - P^n = R_c T^n (\rho^{n+1} - \rho^n) + R_c \rho^n (T^{n+1} - T^n). \quad (1.25)$$

Solving equation (1.25) for the incremental change in density across a time step,

$$\rho^{n+1} - \rho^n = \frac{P^{n+1} - P^n}{R_c T^n} - \frac{\rho^n (T^{n+1} - T^n)}{T^n}, \quad (1.26)$$

and substituting into equation (1.23) to eliminate the new time density ρ^{n+1} as an unknown, yields

$$\frac{P^{n+1}}{R_c T^n} - \Delta t^2 \vec{\nabla} \cdot \vec{\nabla} P^{n+1} = -\Delta t \vec{\nabla} \cdot (\rho \vec{u}^n + \vec{S}) + \frac{P^n}{R_c T^n} + \frac{\rho^n (T^{n+1} - T^n)}{T^n}. \quad (1.27)$$

Equation (1.27) is a Poisson equation in terms of the advanced-time pressure. Note, however, that a new unknown has been introduced by the equation of state, the advanced-time temperature T^{n+1} . This is the energy contribution that must be accounted for and is the reason why most pressure-based schemes for compressible flow are iterative. Equation (1.27), which is essentially the conservation of mass equation, requires the coupling of the total energy to the density and pressure through the application of the equation of state.

The right-hand side of equation (1.27) can be further simplified by defining a term, similar to \vec{S} in the momentum pressure-correction equation, that contains only known explicit contributions as

$$G = -\Delta t \vec{\nabla} \cdot (\rho \vec{u}^n + \vec{S}). \quad (1.28)$$

G in this case is fairly simple. A more complicated G is contained in the PCICE-FEM formulation. It contains time-weighted mass flux terms due to the second-order temporal discretization and an old time pressure Laplacian term due to the change in pressure form of the Poisson equation in PCICE-FEM. G is also held constant in time for a given time step in an iterative scheme. Substituting G into equation (1.27) and simplifying the equation of state terms on the right-hand side gives the final form of the pressure Poisson equation for this basic scheme,

$$\frac{P^{n+1}}{R_c T^n} - \Delta t^2 \vec{\nabla} \cdot \vec{\nabla} P^{n+1} = G + \frac{\rho^n T^{n+1}}{T^n}. \quad (1.29)$$

A typical solution algorithm for this basic mass-momentum coupled pressure-based scheme would have the following iterative steps:

- 1) Compute \vec{S} and then G according to equations (1.22) and (1.28), respectively.
- 2) Estimate an initial T^{n+1} . For example, initially set $T^{n+1} = T^n$.
- 3) Solve the pressure Poisson equation (1.29) for P^{n+1} .
- 4) Update the pressure-corrected momentum $\rho \vec{u}^{n+1}$ employing equation (1.21).
- 5) Compute a new time density ρ^{n+1} based upon the equation of state, equation (1.26).
- 6) Determine a new time total energy ρe_t^{n+1} from equation (1.20) using the updated values of ρ^{n+1} , $\rho \vec{u}^{n+1}$, and P^{n+1} .
- 7) Check for convergence on P^{n+1} . Compute a new T^{n+1} from equation (1.12). If P^{n+1} is not converged, and repeat steps 3 through 7. If P^{n+1} is converged, then go to next time step.

1.2.2 Energy-Momentum Coupling

Energy-momentum coupling may be the preferred coupling technique for flow fields that are strongly energy dependent. Where mass-momentum coupling sought to satisfy mass conservation requirements, energy-momentum coupling results in a pressure field that satisfies conservation of energy requirements. In this case however, the implicit variables are momentum, pressure, and total energy. As the name energy-momentum coupling implies, the momentum balance equations are substituted into the conservation of total energy equation to eliminate momentum as an unknown. This results in a pressure Poisson equation that is essentially a conservation of total energy equation in terms of an implicit pressure and total energy. The equation of state replaces the total energy in terms of pressure. The mass conservation equation must then be coupled to the pressure Poisson equation in an iterative fashion to satisfy mass conservation requirements.

An energy-momentum coupling technique is more complicated than mass-momentum coupling for equation systems in conservation form because the convective fluxes for conservation of total energy, found in equation (1.2), contain the stagnation enthalpy H as well as the components of momentum $\rho \vec{u}$. The stagnation enthalpy H , defined by equation (1.6), is a function of implicit pressure and total energy and time-advanced density in the temporal discretized formulation. There is no way to simultaneously eliminate both time-advanced total energy and the time-advanced momentum components as unknowns in the convective flux term of the conservation of total energy equation. This results in a pressure Poisson formulation that would be difficult at best to solve numerically. The simplest way around this problem is to define H explicitly. The basic derivation presented here will employ this simplified approach by redefining the temporal discretization of equation (1.20) as

$$\rho e_t^{n+1} - \rho e_t^n = -\Delta t \vec{\nabla} \cdot (\rho \vec{u}^{n+1} H^n) \quad (1.30)$$

where, by equation (1.6),

$$H^n = \frac{\rho e_t^n + P^n}{\rho^n}.$$

Proceeding with the energy-momentum coupling, substitute equation (1.21) into equation (1.30) to yield

$$\rho e_t^{n+1} - \rho e_t^n = -\Delta t \vec{\nabla} \cdot \left[(\rho \vec{u}^n + \vec{S}) H^n \right] + \Delta t^2 \vec{\nabla} \cdot (H^n \vec{\nabla} P^{n+1}). \quad (1.31)$$

Equation (1.31) is a pressure Poisson equation in terms of the new time pressure and explicit stagnation enthalpy with the explicit energy flux term and the unknown increment in total energy serving as source terms. This equation contains two unknowns, ρe_t^{n+1} and P^{n+1} , for which the equation of state will be used to relate total energy in terms of pressure. For this basic energy-momentum formulation, equation (1.11) is an equation of state that relates

pressure in terms of total energy. In terms of the conserved variables, the forward-time discretized equation of state is

$$\frac{P^{n+1} - P^n}{\gamma - 1} = \rho e_t^{n+1} - \rho e_t^n - \left(\frac{\rho \vec{u} \cdot \rho \vec{u}}{2\rho} \right)^{n+1} + \left(\frac{\rho \vec{u} \cdot \rho \vec{u}}{2\rho} \right)^n. \quad (1.32)$$

Solving equation (1.32) for the incremental change in total energy and substituting for the left-hand side of equation (1.31) yields

$$\begin{aligned} \frac{P^{n+1}}{\gamma - 1} - \Delta t^2 \vec{\nabla} \cdot (H^n \vec{\nabla} P^{n+1}) &= \frac{P^n}{\gamma - 1} - \Delta t \vec{\nabla} \cdot [(\rho \vec{u}^n + \vec{S}) H^n] \\ &+ \left(\frac{\rho \vec{u} \cdot \rho \vec{u}}{2\rho} \right)^{n+1} - \left(\frac{\rho \vec{u} \cdot \rho \vec{u}}{2\rho} \right)^n. \end{aligned} \quad (1.33)$$

As before with the mass-momentum coupled formulation, the explicit terms, that are constant in time for an iterative scheme, can be grouped together. Defining E as

$$E = \frac{P^n}{\gamma - 1} - \Delta t \vec{\nabla} \cdot [(\rho \vec{u}^n + \vec{S}) H^n] - \left(\frac{\rho \vec{u} \cdot \rho \vec{u}}{2\rho} \right)^n \quad (1.34)$$

and substituting into equation (1.33) yields the final form of the energy-momentum coupled pressure Poisson equation,

$$\frac{P^{n+1}}{\gamma - 1} - \Delta t^2 \vec{\nabla} \cdot (H^n \vec{\nabla} P^{n+1}) = E + \left(\frac{\rho \vec{u} \cdot \rho \vec{u}}{2\rho} \right)^{n+1}. \quad (1.35)$$

Notice that the right-hand side of equation (1.35) requires implicit values of momentum and density, $\rho \vec{u}^{n+1}$ and ρ^{n+1} , respectively. Therefore, $\rho \vec{u}^{n+1}$ and ρ^{n+1} will have to be updated in an iterative loop. Conceivably, stagnation enthalpy H^n , in terms of ρ^{n+1} and P^{n+1} , could be included in the iterative loop. However, for this basic scheme, H^n is left explicit. The iterative algorithm for the solution of this basic energy-momentum pressure-based scheme is as follows:

- 1) Compute \vec{S} and then E according to equations (1.22) and (1.34), respectively.
- 2) Estimate an initial $\rho \vec{u}^{n+1}$ and ρ^{n+1} for the right-hand side of equation (1.35).
- 3) Solve the pressure Poisson equation (1.35) for P^{n+1} .
- 4) Update the pressure-corrected momentum $\rho \vec{u}^{n+1}$ with equation (1.21).
- 5) Compute a new time density ρ^{n+1} with equation (1.17).

- 6) Determine a new time total energy ρe_t^{n+1} from the equation of state, equation (1.32).
- 7) Check for convergence on P^{n+1} . If P^{n+1} is not converged, then repeat steps 3 through 7. If P^{n+1} is converged, then go to the next time step.

1.2.3 Previous Semi-Implicit Pressure-Based Research Efforts

At the time of this writing, the literature contains a large number of references pertaining to semi-implicit pressure-based schemes. There are too many to fully review here. However, a good representation of previous research efforts can be obtained by reviewing the most widely varying approaches into this topic.

It is realistic to say that the birth of computational fluid dynamics (CFD) occurred at the Los Alamos National Laboratory (LANL) during the Manhattan Project. In the late 1950s, the world's largest computer resources were located at LANL for research into CFD methods. Programs of national security led to the formation of the Fluid Dynamics Group (T-3) in the Theoretical Division at LANL in 1958 with Francis H. Harlow as its first Group Leader (Johnson, 1996). The main problems of interest for T-3 were concerned mainly with multiple materials under high compression in which solids behave like fluids. It was at this time that the Particle in Cell (PIC) method was proposed and developed by Harlow in 1957 (Evans and Harlow, 1957). The original PIC method used mass particles that carried material position, mass, and species information on a two-dimensional uniform Eulerian grid. It was used to simulate transient compressible flows with multiple materials. To treat incompressible free surface flows, the Marker-And-Cell (MAC) method was developed by Harlow and Welch (1965a and 1965b) as a variation of the PIC method. The MAC scheme was the first successful technique for incompressible flows. In order to treat fluid incompressibility, the MAC method incorporated a solution to a Poisson equation for pressure.

Resulting from this early research at LANL, the Implicit Continuous-fluid Eulerian (ICE) scheme was developed by Harlow and Amsden (1968 and 1971). While PIC and later the FLuid-In-Cell (FLIC) scheme (Gentry et al., 1966) were developed to solve compressible flow problems and the MAC method addressed incompressible flows, the ICE scheme was developed as an "all speed" method. The ICE scheme was the first approach that removed the acoustic component from the Courant stability limitation for maximum time step size. For nearly incompressible flows (the sound speed approaching infinity), the ICE scheme essentially reduces to the MAC scheme. The ICE scheme has served as the basis for a number of computer programs developed at LANL. SOLA-ICE (Cloutman et al., 1976) combined the SOLA scheme (Hirt et al., 1975), which is a simplified MAC scheme, with the ICE method for transient compressible or incompressible flows. RICE (Rivard et al., 1975) is a version of ICE developed to simulate chemically reacting flows. The RICE scheme was the basis for the APACHE-CONCHAS-KIVA series of multi-phase chemically reacting flow codes (Ramshaw and Dukowicz, 1979; Dukowicz, 1980; Amsden et al., 1989; and Amsden, 1993). The ICED-ALE scheme, developed by Amsden and Hirt (1973), was the combination of ICE with the Arbitrary-Lagrangean-Eulerian (ALE) method (Hirt et al., 1974). A modern version of the

ICED-ALE scheme has been incorporated into LANL's CFDLIB collection of codes (Kashiwa et al., 1994).

For Harlow and Asmden's original ICE scheme, the mass conservation and the balance of momentum equations are coupled together in the pressure Poisson. A barotropic like equation of state is employed to relate the new time pressure to the new time density. After the new time pressure is obtained from the pressure Poisson equation, the new time density is determined from the equation of state and the new time balance of momentum components are determined by the momentum pressure-correction equations. The new time pressure, momentum, and density are then used to solve the energy conservation equation in an explicit fashion. The energy conservation equation is, therefore, decoupled from the mass conservation and momentum balance equations. In other words, new time energy information is not required to solve the governing system of hydrodynamic equations. A barotropic like equation of state (for example, equation (1.24) with the second term on the right-hand side removed) assumes that the variations in pressure are due mainly to the variations in density. This assumption is valid for problems where the variation in internal energy is small or energy deposition rates are minimal. If the change in internal energy is small, then the correction to the pressure field through the full equation of state is also small. The barotropic assumption incorporated into the original ICE scheme does not imply that the variations in internal energy can be ignored. It simply means that the variations in pressure are treated implicitly with respect to density and explicitly with respect to internal energy. However, the barotropic fluid assumption causes problems for the original ICE scheme simulating transient flows that are highly compressible, chemically reactive, or are subjected to high heat fluxes. Because the original ICE scheme is non-iterative, flow features, such as shocks and contact discontinuities, that require strong coupling of the energy conservation equation are not resolved properly. For these types of flow, the ICE method also fails to completely eliminate the acoustic component of the Courant stability restriction. Westbrook (1978) addresses these problems with the development of a generalized ICE scheme for combustion simulations. This generalization of the equation of state allows for the pressure dependency upon internal energy and chemical kinetics to be taken into account. This requires an iterative ICE scheme where the advanced-time pressure obtained from the pressure Poisson equation is compared with the advanced-time pressure computed from the equation of state to determine convergence.

The semi-implicit method for pressure-linked equations (SIMPLE) scheme (Patankar and Spalding, 1972) and its variants SIMPLER (Patankar, 1980), SIMPLEC (Van Doormaal and Raithby, 1984), SIMPLEX (Van Doormaal and Raithby, 1985), and SIMPLEST (Sha, 1985) have proven to be popular pressure-based algorithms for solving incompressible flows. Van Doormaal et al. (1987) and Bai et al. (1987) have extended the SIMPLE algorithms to handle compressible flows. The SIMPLE algorithms are formulated in terms of a pressure-correction variable, the difference between "predicted" pressure and "corrected" pressure. An iterative loop is employed between the pressure Poisson equation and the momentum balance equations, both in terms of the pressure-correction variable. Convergence is achieved when the pressure-correction is globally near zero. For compressible flows, this loop constitutes an "inner" loop while an "outer" loop is required to account for energy effects. An improved method, termed the pressure-implicit with splitting of operators (PISO) and developed by Issa (1985) and Issa et al. (1991), was initially derived for the accurate simulation of transient

flows. PISO is a non-iterative scheme requiring a single predictor stage and two corrector stages. The predictor stage is composed of an explicit step for momentum with a corresponding pressure Poisson solution to obtain a new time predicted pressure field which results in a new density field obtained from the equation of state. This is followed by the first corrector step for momentum. The corrected momentum values are used to update the energy conservation equation and equation of state variables. The pressure Poisson is again solved to obtain a corrected mass-balance pressure field from which a new density distribution is obtained. A final update is then performed for the momentum balance and energy conservation equations. The PISO and SIMPLE schemes are often combined in the same computer code because of they employ the same staggered finite volume discretization and the similarity in the pressure correction formulations. PISO is generally more accurate in transient compressible flow simulations while SIMPLE is more efficient for incompressible and steady-state simulations.

An institution that specializes in finite element methods is the Institute of Numerical Methods in Engineering, College of Swansea, United Kingdom. It is here that one of the most (if not the most) prolific publishers of finite element methods, Professor O.C. Zienkiewicz, resides. Throughout the 1990's, Professor Zienkiewicz and a series of graduate students developed a general explicit or semi-implicit algorithm for compressible and incompressible flows (Zienkiewicz and Wu, 1991 and 1992; Zienkiewicz and Codina, 1995; Zienkiewicz et al., 1995; Codina et al., 1998; and Zienkiewicz et al., 1999) called characteristic-based-split (CBS) algorithm. In its semi-implicit form, this algorithm is similar to that of the present PCICE-FEM scheme presented here. The CBS algorithm is also composed of three phases, an explicit predictor, an elliptic pressure Poisson solution, and an implicit pressure-correction of the flow variables. The explicit predictor for CBS is based upon a characteristic Galerkin scheme that is similar to the one-step Taylor-Galerkin scheme (Löhner et al., 1984). It incorporates a time-weighted fractional step momentum equation which predicts momentum with out the influence of pressure. The resulting momentum pressure-correction equation found in the CBS algorithm is identical to that found in the PCICE-FEM scheme and the pressure Poisson equation for both schemes is nearly identical except for the application of boundary conditions (see Chapter 3). The main drawback of the CBS algorithm is the fully explicit treatment of the energy conservation equation. The total energy is never updated with the new time pressure, momentum, and density. This procedure works well for steady-state flows where the change in flow variables is ultimately zero. For high speed compressible flows, the CBS algorithm requires iteration between the pressure Poisson equation and the implicit updates to obtain the correct temperature distribution to satisfy the equation of state. Another point of observation, the CBS discretization is derived second-order accurate in time. However, in none of the above references is the CBS algorithm demonstrated on transient flow problems. If the iteration process is not performed, the order of approximation is reduced to first-order in time (Codina et al., 1998) which is sufficient for steady-state solutions. The chosen procedure for the CBS algorithm is to perform at least two iterations, regardless of the problem type, in order to slightly increase the maximum allowable time step size.

While most pressure-based schemes are developed from mass-momentum coupling, the energy-momentum coupling technique has two important advantages. Energy-momentum coupling strongly satisfies energy conservation which is important for flows that have strong

energy dependence. Another important advantage of energy-momentum coupling is that it allows for a more general treatment of the equation of state for compressible flows in the pressure Poisson equation than found in the original ICE scheme. The characteristic analysis performed by Casulli and Greenspan (1984) indicates that the pressure gradient terms in the momentum balance equations and the divergence of velocity term in the energy conservation equation should be discretized implicitly to fully remove the acoustic component from the Courant time step limitation. The resulting pressure method from Casulli and Greenspan's research is the basis of most energy-momentum coupled pressure-based formulations. Casulli and Greenspan employed a primitive variable form of the governing equations which simplifies the energy-momentum coupling formulation. Patnaik et al. (1987) extended Casulli and Greenspan's pressure method, in conservative variable form, to remove the acoustic component from the Courant condition for an explicit FCT finite difference method. The resulting scheme is named BIC-FCT for barely implicit correction. Both Casulli and Greenspan's pressure method and the BIC-FCT scheme treat the mass conservation equation in an explicit manner, which effectively decouples the mass conservation equation. Both Casulli and Greenspan's and BIC-FCT pressure methods treatment of the mass conservation equation is strictly explicit where mass is convected with old time information only. This brings into question BIC-FCT's ability to adequately simulate nearly incompressible flows. In reality though, energy-momentum coupling is best suited to flows that are strongly dependent upon energy.

A novel new method to simultaneously simulate compressible and incompressible flows is the Combined Unified Procedure (CUP). This algorithm was first forwarded by Yabe and Wang (1991) to provide a universal solver to simulate different materials and phases using the same scheme and governing equations. The CUP method has since been modified to include the simulation of immersed solid bodies (moving or stationary) and plasmas (Xiao and Yabe, 1994; Yabe et al., 1995 and 1999; and Xiao et al., 1997). This algorithm is essentially an energy-momentum coupled pressure-based method that also satisfies mass conservation requirements. Thus, materials with widely varying sound speeds can be easily incorporated into the pressure equation with only the material velocity determining a stable time step increment. In the current version of CUP (Yoon and Yabe, 1999), a predictor-corrector method is employed to correctly calculate the pressure. The governing equations are separated into advection and non-advection phases by a time-splitting method similar to Chorin's (1967). The advection phase is simulated by the Cubic-Interpolated Propagation (CIP) scheme (Yabe and Wang, 1991) with spatial accuracy of third-order. The non-advection phase is discretized by a center-space finite difference method with second-order accuracy. Because CIP uses a primitive variable formulation of the governing equations, the scheme is robust and accurate at material boundaries that have large differences in density, such as the interface between air and water. Conservative variable formulations tend to be more diffusive at material interfaces. The combination of CUP and CIP, called CCUP (Yoon and Yabe, 1999), has shown impressive results for simulating domains where gas, liquid, and solid phases are all present.

1.3 Specific Objectives of this Research

The goal of any new research effort in computational methods is to extend beyond the current state-of-the-art methods. Here, the overall goal is to develop a superior semi-implicit, mass-momentum coupled, pressure-based algorithm for compressible flows that significantly improves upon previous research efforts in terms of accuracy, consistency of formulation, and numerical efficiency. This goal has been accomplished with the development of the PCICE-FEM scheme. In general terms, this scheme can be described as an ICE formulation of an unstructured FEM spatial discretization that incorporates FCT as a high-resolution filter. The specific goals that led to the development of the PCICE-FEM scheme are as follows:

- 1) *For maximum efficiency, the scheme must be non-iterative, matrix free, and exactly integrated:* In order for the scheme to be non-iterative, sufficient inclusion of the energy effects with the mass-momentum coupling must be obtained. This is accomplished in the PCICE-FEM scheme by incorporating an explicit total energy predictor and a semi-implicit total energy corrector. Unstructured finite element meshes can generate matrices with large global bandwidths and result in slow numerical operation. Numbering the nodes and elements to minimize memory requirements and cache misses is essential for large simulations. Also, the matrices can be avoided altogether if the scheme can be formulated in terms of matrix-vector dot products. Exact integration of unstructured linear finite elements allows easy assembly of the matrix-vector dot product contributions while avoiding the heavy computational cost of numerical integration.
- 2) *Scheme should be able to utilize any desired equation of state:* For the scheme to have the widest range applicability, the scheme must not be restricted, by formulation, to a single equation of state. The only requirement on the equation of state that is incorporated into the PCICE-FEM scheme is that the density must be represented in terms of pressure.
- 3) *Incorporate Flux-Corrected Transport as the high-resolution filter:* FCT has shown excellent shock capturing abilities for transient simulations with discontinuities in the flow variables being captured within three elements for multi-dimensional domains. However, FCT does not appear to introduce enough artificial dissipation to achieve complete convergence for steady-state problems. Steady-state must be qualitatively determined. FCT was originally developed for purely explicit schemes. It is also used in this fashion for the PCICE-FEM scheme where FCT is used as a filter on the explicit predictor phase to control numerical oscillations in the flow variables generated by the second-order spatial discretization.
- 4) *Boundary conditions must be robust and accurate:* The governing equations of compressible flow define a system of nonlinear hyperbolic first-order partial differential equations. In the case of a finite spatial domain, the system of governing equations constitute an initial-boundary value problem. Mathematically correct initial conditions and boundary conditions are essential in defining a well-posed problem. Application of boundary conditions are dictated by the characteristic form of the

governing equations. They dictate what may or may not be specified at the boundaries of the domain. Rigorous mathematical formulation and discretization of the boundary conditions are required to ensure robust and accurate behavior of the simulation.

- 5) *Steady-state acceleration technique must be incorporated:* A second-order semi-implicit formulation was chosen over fully implicit schemes for several reasons. While fully implicit schemes theoretically have no stability requirement and thus can be very efficient for steady-state simulations, they require considerably more complex algorithms which generally results in a much larger computational cost per time step. This is a liability for computing transient simulations where the time step must be small enough to accurately resolve the transient phenomena. This requirement dictates that the time step must be smaller than or equal to the material Courant limit which is the time step size limit of semi-implicit pressure-based schemes. Because semi-implicit schemes require much less computational effort per time step than fully implicit schemes, they are usually more efficient for transient applications. Though semi-implicit schemes are limited by the material Courant condition for steady-state applications, steady-state acceleration techniques can and should be incorporated into semi-implicit formulations to significantly reduce computational run times for steady-state simulations. Residual smoothing is a second-order differential effect incorporated into the PCICE-FEM scheme that increases the level of implicitness of the formulation. Being elliptic in nature, residual smoothing causes transients to have a wider influence upon the domain than the physics dictate. Thus, the simulation is marched through a false transient. As the solution nears steady-state, the amount of residual smoothing decreases to zero and the actual steady-state solution is recovered.

2. The Explicit Two-Step Taylor-Galerkin Finite Element Method for Unstructured Grids (FEM-FCT)

Application of the finite element method (FEM) to the solution of high-speed compressible flow problems has been extensively researched by Löhner et al., (1985-1988). Successful implementation of the two-step Taylor-Galerkin FEM for systems of hyperbolic equations resulted in solutions as accurate as those obtained from more traditional methods (finite difference and finite volume methods). Originally, the Taylor-Galerkin FEM employed by Löhner (1985a) for modeling systems of hyperbolic equations required the addition of artificial viscosity to stabilize the solution procedure in areas involving strong discontinuities; i.e., shocks and blast waves. This artificial viscosity effectively diffuses the flow discontinuities over several computational cells. Therefore, high-resolution schemes which would give sharper definition of flow discontinuities were needed. Zalesak (1979) developed a multi-dimensional generalization of the one-dimensional Flux-Corrected Transport (FCT) solution scheme, developed by Boris and Book (1973-1976), which is directly extendible to produce high resolution schemes on unstructured grids. This method combines a high-order scheme together with a low-order scheme in such a way that the high-order flux is used in regions where the solution is smooth. In regions where the solution contains high changes in gradient or discontinuities, the high and low-order schemes are combined and limited so that the solution is monotonic and conservative. Löhner (1987 and 1988) implemented this generalized concept of FCT into the Taylor-Galerkin FEM for systems of hyperbolic equations. This solution method is known as the Finite Element Flux-Corrected Transport (FEM-FCT) scheme.

The high-order solution for the FEM-FCT scheme is obtained by a two-step form of the Taylor-Galerkin scheme developed by Donea (1984). The Taylor-Galerkin scheme is used to increase the order of the approximation of the time derivative to second-order. This is accomplished by expanding the time derivative of a Taylor series to second-order and replacing the time derivatives with the spatial derivatives of the governing hydrodynamic equations. This second-order time discretization was also found to improve phase accuracy and minimize dissipation errors (Donea, 1984). The spatial discretization of the high-order scheme is formulated with the Galerkin weighted residual method. The scheme incorporates two steps which together advance the solution from time t^n to $t^{n+1} = t^n + \Delta t$. The first step advances the solution to the cell centers at time $t^{n+\frac{1}{2}} = t^n + \frac{\Delta t}{2}$. This half-step solution provides fluxes for the second step which advances the solution to the end of the time step at time t^{n+1} . Solution variables at the beginning and ending of a time step (t^n and t^{n+1}) are located at the nodes of the element and are interpolated by piecewise linear shape functions. At the half-step ($t^{n+\frac{1}{2}}$), variables are given at the element centroids and are interpolated by piecewise constant shape functions. The FEM-FCT scheme was developed primarily for computational fluid dynamics (CFD) problems on unstructured grids. Linear triangular elements (two-dimensional) and linear tetrahedral elements (three-dimensional) are employed for several reasons. The first is that they are easily generated on domains with complex geometries. Second, they provide second-order spatial accuracy. Third, they can be easily adapted to minimize error in the solution. And lastly, they can be integrated exactly, which precludes the need of time consuming numerical integration.

The FEM-FCT scheme produces excellent results for the simulation of transient compressible flows containing shocks and contact discontinuities. It is also a superior scheme for simulating high-speed steady-state inviscid flows where the entire flow field is supersonic. As with most purely explicit schemes, the stability of the FEM-FCT scheme is governed by the Courant-Friedrichs-Lewy, or CFL, condition. The CFL condition dictates the largest allowable time step that may be applied to achieve a stable solution. For most explicit schemes, the CFL condition takes the form of

$$\Delta t \leq \frac{\alpha \beta h_e}{|\vec{u}| + c} \quad (2.1)$$

where h_e is the element height, $|\vec{u}|$ is the magnitude of the local velocity, c is the local sound speed, α is a dimensionless parameter that is determined by a linear stability analysis, and β is a user defined factor of safety. The requirement of c in the denominator of equation (2.1) tends to limit the applicability of explicit schemes for domains containing regions where the flow is much slower than the Mach number. This is especially true for viscous flows where the boundary layer discretization contains elements with very small element heights while the sound speed may be very high. This condition results in a maximum allowable time step size that is too restrictive to efficiently simulate steady-state viscous flows by explicit methods. Addressing this problem is one of the main reasons for developing the PCICE-FEM scheme. Semi-implicit, mass-momentum coupled, pressure-based schemes do not have the acoustic component, c , in the CFL condition. Equation (2.1) without the acoustic component is commonly referred to as the material Courant condition because the denominator contains only the material velocity component. For boundary layer discretizations, the allowable time step is nearly infinite because the velocity magnitude may be nearly zero.

As discussed in Chapter 1, the FEM-FCT scheme is the basis for the explicit predictor phase of the PCICE-FEM scheme. Full derivation of the FEM-FCT scheme will be illustrated in this chapter for three main reasons. First, the consistent-mass two-step Taylor-Galerkin FEM (the high-order accurate scheme) needs to be illustrated in order to show how this scheme is modified for the explicit predictor phase of the PCICE-FEM scheme. Second, the application of FCT as a high-resolution filter for the explicit predictor of the PCICE-FEM scheme is identical to that found in the FEM-FCT scheme. And third, a rigorous application of characteristic theory for boundary conditions has been developed for the FEM-FCT scheme. This new application of characteristic boundary conditions is consistent with the spatial and temporal discretization of the Two-Step Taylor-Galerkin FEM scheme. This new characteristic boundary condition formulation has proven to be very robust and accurate for transient and steady-state compressible flow simulations. This chapter will end with simulation examples demonstrating the robustness of the new characteristic boundary conditions.

2.1 The High-Order Scheme: The Consistent-Mass Two-Step Taylor-Galerkin FEM

The high-order solution incorporated into the FEM-FCT scheme employs a two-step form of the Taylor-Galerkin scheme as described by Donea (1984) and Löhner et al., (1985a). This scheme is a combination of the two-step Lax-Wendroff temporal integration method and the FEM for spatial integration. The first step of the two-step Taylor-Galerkin scheme is an advective predictor (ignoring the viscous flux vector, \vec{F}_v) designed to advance the solution to the half-step, denoted $t = t^{n+\frac{1}{2}}$. This is accomplished by writing a Taylor's series for U about time $t + \Delta \frac{\Delta t}{2}$, yielding

$$U^{n+\frac{1}{2}} = U^n + \frac{\Delta t}{2} \frac{\partial U^n}{\partial t} + O(\Delta t)^2. \quad (2.2)$$

Substituting equation (1.2), without the viscous terms, into equation (2.2), yields

$$U^{n+\frac{1}{2}} = U^n - \frac{\Delta t}{2} \vec{\nabla} \cdot \vec{F}^n + \frac{\Delta t}{2} Q^n. \quad (2.3)$$

Denoting the piecewise linear shape function associated with node j by N_j , which will be used for terms at time t^n and t^{n+1} , and the piecewise constant shape function associated with element e by P_e , which will be used for terms at time $t = t^{n+\frac{1}{2}}$, the finite element approximations for the first-step of the Taylor-Galerkin method are

$$U^n \approx \sum_j U_j^n N_j = [U]^n \{N\}, \quad U^{n+\frac{1}{2}} \approx \sum_e U_e^{n+\frac{1}{2}} P_e, \quad \vec{F}^n \approx \sum_j \vec{F}_j^n N_j = [\vec{F}]^n \{N\},$$

$$\text{and } Q^n = \sum_e Q_e^n P_e$$

where $P_e = 1$ on element e and $P_e = 0$ over the rest of the domain. The source terms, Q , require piecewise constant representation because they are defined only on the elemental level. A column matrix is denoted by $\{ \}$ and any other matrix is denoted by $[]$. U_j and \vec{F}_j are nodal values of U and \vec{F} , respectively. $U_e^{n+\frac{1}{2}}$ are elemental values of U . The sums over j and e are the sums over all the nodes and elements in the domain, respectively. Substituting these approximations into the weighted residual form of equation (2.3) and integrating over the domain, yields

$$\sum_e \int_{\Omega} U_e^{n+\frac{1}{2}} P_e W d\Omega = \int_{\Omega} [U]^n \{N\} W d\Omega - \frac{\Delta t}{2} \int_{\Omega} [\vec{F}]^n \cdot \{\vec{\nabla} N\} W d\Omega + \frac{\Delta t}{2} \sum_e \int_{\Omega} Q_e^n P_e W d\Omega \quad (2.4)$$

where Ω is the domain. Noting that equation (2.4) is solved for elemental values of U at time $t = t^{n+\frac{1}{2}}$, the Galerkin approximation is piecewise constant, $W = P_E$, where E corresponds to all of the elements in the domain. Equation (2.4) then becomes

$$\sum_e \int_{\Omega} U_e^{n+\frac{1}{2}} P_e P_E d\Omega = \int_{\Omega} [U]^n \{N\} P_E d\Omega - \frac{\Delta t}{2} \int_{\Omega} [\vec{F}]^n \cdot \{\vec{\nabla} N\} P_E d\Omega + \frac{\Delta t}{2} \sum_e \int_{\Omega} Q_e^n P_e P_E d\Omega \quad (2.5)$$

for every E . Using the fact that piecewise constant shape functions are nonzero only on a given element, equation (2.5) can be integrated over element E to yield the final form of the first step,

$$\Omega_E U_E^{n+\frac{1}{2}} = \int_{\Omega} [N] d\Omega \{U\}^n - \frac{\Delta t}{2} \int_{\Omega} [\vec{\nabla} \cdot N] d\Omega \{\vec{F}\}^n + \frac{\Delta t}{2} \Omega_E Q^n \quad (2.6)$$

where Ω_E is the volume of element E . Note that the right hand side of equation (2.6) requires integration only over element E versus integration over all elements with a common node and is formally a finite volume method. The assembly process that is usually associated with the finite element method is then not necessary. Equation (2.6) closely resembles the Lax-Friedrichs (Lax, 1954) finite difference approximation of equation system (1.2).

The second step of the two-step Taylor-Galerkin scheme is developed by writing Taylor series for U^n about time $t^{n+\frac{1}{2}}$ and U^{n+1} about time $t^{n-\frac{1}{2}}$. Subtracting these two series gives

$$U^{n+1} = U^n + \Delta t \frac{\partial U^{n+\frac{1}{2}}}{\partial t} + O(\Delta t)^3. \quad (2.7)$$

Substituting equation (1.2) into equation (2.7), yields the second step of the Taylor-Galerkin scheme

$$U^{n+1} = U^n - \Delta t \vec{\nabla} \cdot \vec{F}^{n+\frac{1}{2}} + \Delta t \vec{\nabla} \cdot \vec{F}_v^n + \Delta t Q^{n+\frac{1}{2}}. \quad (2.8)$$

By letting $\delta U = U^{n+1} - U^n$, the weighted residual form of equation (2.8) is

$$\int_{\Omega} \delta U W d\Omega = -\Delta t \int_{\Omega} \vec{\nabla} \cdot \vec{F}^{n+\frac{1}{2}} W d\Omega + \Delta t \int_{\Omega} \vec{\nabla} \cdot \vec{F}_v^n W d\Omega + \Delta t \int_{\Omega} Q^{n+\frac{1}{2}} W d\Omega. \quad (2.9)$$

The finite element approximations for equation (2.9) are

$$U^{n+1} \approx \sum_j U_j^{n+1} N_j = [U]^{n+1} \{N\}, \quad U^n \approx \sum_j U_j^n N_j = [U]^n \{N\}, \quad \vec{F}^{n+\frac{1}{2}} \approx \sum_e \vec{F}_e^{n+\frac{1}{2}} P_e, \\ F_v^n \approx \sum_e F_v^n P_e, \quad \text{and} \quad Q^{n+\frac{1}{2}} = \sum_e Q_e^{n+\frac{1}{2}} P_e.$$

$\vec{F}_e^{n+\frac{1}{2}}$ is the convective flux vector at time $t^{n+\frac{1}{2}}$ and is constructed from elemental values of $U_e^{n+\frac{1}{2}}$, the half-step solution. \vec{F}_v contains first-order derivatives which are constants when defined across a linear finite element. Evaluating \vec{F}_v from the half-step solution, $U_e^{n+\frac{1}{2}}$, which is constant on an element, is difficult. Therefore, \vec{F}_v is lagged to time t^n so the first derivatives can be evaluated from nodal values.

Applying the standard Galerkin approximation, $W = N_j$, and substituting the above finite element approximations into equation (2.9), yields

$$\begin{aligned} \sum_e \int_{\Omega} [\delta U] \{N\} N_j d\Omega = & -\Delta t \sum_e \int_{\Omega} \vec{\nabla} \cdot (\vec{F}_e^{n+\frac{1}{2}} P_e) N_j d\Omega + \Delta t \sum_e \int_{\Omega} \vec{\nabla} \cdot (\vec{F}_v^n P_e) N_j d\Omega \\ & + \Delta t \sum_e \int_{\Omega} Q_e^{n+\frac{1}{2}} P_e N_j d\Omega \end{aligned} \quad (2.10)$$

for each N_j . The first two integrals on the right hand side of equation (2.10) contain the derivative of a piecewise constant, making for awkward evaluation. Integration by parts is therefore applied to these integrals. This shifts part of the differentiability requirement from $\vec{F}_e^{n+\frac{1}{2}} P_e$ and $\vec{F}_v^n P_e$ to N_j . Applying Green's theorem (integration by parts) and completing the matrix operations, equation (2.10) becomes

$$\begin{aligned} \sum_e \int_{\Omega} \{N\} [N] d\Omega \{\delta U\} = & \Delta t \sum_e \int_{\Omega} P_e \{\vec{\nabla} N\} d\Omega \cdot \vec{F}_e^{n+\frac{1}{2}} - \Delta t \sum_e \int_{\Gamma} P_e \{N\} d\Gamma \vec{\eta} \cdot \vec{F}_e^{n+\frac{1}{2}} \\ & - \Delta t \sum_e \int_{\Omega} P_e \{\vec{\nabla} N\} d\Omega \cdot \vec{F}_v^n + \Delta t \sum_e \int_{\Gamma} P_e \{N\} d\Gamma \vec{\eta} \cdot \vec{F}_v^n + \Delta t \sum_e \int_{\Omega} P_e \{N\} d\Omega Q_e^{n+\frac{1}{2}}, \end{aligned} \quad (2.11)$$

where Γ defines the boundary of the domain and $\vec{\eta}$ is the unit vector normal to the boundary. Evaluating the boundary integrals will be discussed later in Section 2.5 when covering the boundary conditions.

Using the fact that P_e is nonzero only on element e and writing the domain integrals as elemental integrals, equation (2.11) becomes the final form of the second step,

$$\begin{aligned} \sum_e \int_{\Omega} \{N\} [N] d\Omega \{\delta U\} = & \Delta t \sum_e \int_{\Omega} \{\vec{\nabla} N\} d\Omega \cdot \vec{F}_e^{n+\frac{1}{2}} - \Delta t \sum_e \int_{\Gamma} \{N\} d\Gamma \vec{\eta} \cdot \vec{F}_e^{n+\frac{1}{2}} \\ & - \Delta t \sum_e \int_{\Omega} \{\vec{\nabla} N\} d\Omega \cdot \vec{F}_v^n + \Delta t \sum_e \int_{\Gamma} \{N\} d\Gamma \vec{\eta} \cdot \vec{F}_v^n + \Delta t \sum_e \int_{\Omega} \{N\} d\Omega Q_e^{n+\frac{1}{2}}. \end{aligned} \quad (2.12)$$

Equation (2.12) can then be represented in the more compact form

$$M_c \delta U = R \quad (2.13)$$

where M_c denotes the consistent mass matrix and is formed from

$$M_c = \sum_e \int_{\Omega} \{N\} [N] d\Omega. \quad (2.14)$$

In equation (2.13), R is the vector of added element contributions to the nodes. These contributions are defined as

$$\begin{aligned} R = \Delta t \sum_e \int_{\Omega} \{\vec{\nabla} N\} d\Omega \cdot \vec{F}_e^{n+\frac{1}{2}} - \Delta t \sum_e \int_{\Gamma} \{N\} d\Gamma \vec{\eta} \cdot \vec{F}_e^{n+\frac{1}{2}} - \Delta t \sum_e \int_{\Omega} \{\vec{\nabla} N\} d\Omega \cdot \vec{F}_v^n \\ + \Delta t \sum_e \int_{\Gamma} \{N\} d\Gamma \vec{\eta} \cdot \vec{F}_v^n + \Delta t \sum_e \int_{\Omega} \{N\} d\Omega Q_e^{n+\frac{1}{2}}. \end{aligned} \quad (2.15)$$

Because M_c possesses an excellent condition number (diagonally dominant), equation (2.13) can be solved most efficiently with an iterative procedure developed by Donea (1984). Equation (2.13) is recast as

$$M_l \delta U + M_c \delta U = R + M_l \delta U \quad (2.16)$$

where M_l denotes the lumped mass matrix. In iterative form, equation (2.16) becomes

$$M_l \delta U_{i+1} = R + (M_l - M_c) \delta U, \quad i = 1, 2, \dots, Niter \quad (2.17)$$

where i is the iteration count and $Niter$ is the number of iterations. For the initial conditions, ($i = 0$), $\delta U_0 = 0$. Three iterations is usually sufficient for convergence. When δU has converged, the high-order solution for FCT has been achieved. For later use, denote the high-order solution from equation (2.17) as U^h .

2.2 Stability of the Two-Step Taylor-Galerkin Finite Element Method

As with all explicit schemes, the consistent mass Two-Step Taylor-Galerkin scheme is conditionally stable and is subject to the Courant type stability restriction found in equation (2.1). The stability criteria can be approximately determined by performing a Von Neumann stability analysis on a linearized set of discretized governing equations (Hirsch, 1988). Satisfaction of the resulting stability criteria is a necessary condition for stability. However, when the governing equations are nonlinear, as they are here, the stability criteria may not always be sufficient. This is the reason for the user specified safety factor β in equation (2.1). An exact stability analysis cannot be performed upon a nonlinear set of governing equations. A β that will render a simulation to be stable can be highly problem dependent, both for the mesh discretizing the domain and the type of flow being simulated. This is especially true for viscous compressible flows where diffusion (viscous) terms must be taken into account. The linear stability analysis performed by Morgan and Peraire (1987) for the consistent mass Two-

Step Taylor-Galerkin scheme upon a scalar advection equation yielded an $\alpha = 1/\sqrt{3}$. A $\beta = 0.8-0.9$ generally gives a stable and accurate solution with this scheme.

2.3 Convergence Acceleration for Steady-State Simulations

If only steady-state solutions are required, the number of time steps or computational effort required to obtain a converged solution can be significantly reduce by implementing one or more of the following:

- Lump the consistent mass matrix in equation (2.13) (Donea, 1984 and Löhner et al., 1985a). This is equivalent to performing the first iterative step in equation (2.17) only. In this case, the factor α in equation (2.1) can be set equal to 1.
- Perform local time stepping (Morgan and Peraire, 1987). This advances the solution in each individual element with the local maximum allowable time step. Equation (2.1) is applied to each element to obtain an elemental time step Δt_e , which is applied to the first solution step, equation(2.6), to obtain δU_e . For the second step, a nodal time step Δt_i is computed as the average of the elemental time steps surrounding node i . This nodal time step is used to replace Δt in equation (2.12).
- Perform residual smoothing (Jameson et al., 1986). Residual smoothing adds an elliptic character to the iterative solution of the second solution step. Equation (2.17) can be rewritten as

$$M_l \delta U_{i+1} = R + \varepsilon (M_l - M_e) \delta U, \quad i = 1, 2, \dots, N_{iter}, \quad (2.18)$$

where $\varepsilon = 1$ for transient simulations and a small negative number for steady-state simulations. Typically for steady-state simulations, $\varepsilon = -0.1$ but this value is somewhat problem dependent. A negative ε renders the second term on the right-hand side of equation (2.18) diffusive. This elliptic character adds a certain level of implicitness to equation (2.18). The maximum allowable time step can then be slightly increased. Setting $\beta = 1.2$ generally achieves a stable solution with three passes of equation (2.18). Notice that the elliptic effect vanishes at steady-state where $\delta U = 0$.

2.4 Stabilizing the High-Order Scheme with Artificial Dissipation

Requirements of positivity and accuracy of compressible flow quantities can not be mutually achieved with the use of traditional second-order spatially accurate discretized numerical schemes. Godunov (1959) showed that linear second-order or higher-order spatially accurate methods can not guarantee monotonicity. First-order methods, such as the Lax-Friedrichs method are monotonic and guarantee positivity but are extremely diffusive, which tends to smear high gradients and discontinuities over many computational cells, thus rendering the solution inaccurate. Second-order methods, such as the Lax-Wendroff or MacCormack methods, are less diffusive but are susceptible to nonlinear instabilities and nonphysical oscillations occurring in regions where the solution contains high changes in gradient

(curvature) and discontinuities. These oscillations can cause conserved variables, which physically (and computationally) require positivity, to become negative and possibly render the simulation unstable. Oran and Boris (1987) give an excellent description of how these problems occur with low and high-order methods. Some form of artificial dissipation is required in the numerical simulation of compressible flows in order to maintain a stable and physically relevant representation of the flow.

Introducing artificial diffusion into the algorithm is one traditional approach to dampen the oscillations created by high-order methods. However, this diffusion also spreads high gradients over more computational cells as the overshoots and undershoots are suppressed. Total damping of the oscillations requires as much numerical diffusion as first-order methods. Because the addition of artificial dissipation is essentially introducing nonphysical behavior into the flow simulation, it must be carefully applied in such a manner that excessive error is not introduced into the solution and that conservation is strictly maintained. A variable dissipation method is needed that automatically senses where the artificial dissipation should be applied. The method should only apply dissipation in regions where the solution contains high changes in solution gradient, not in regions where the solution is already smooth. Two such artificial dissipation methods are presented here. The first was developed by Lapidus (1967) and later applied to the finite element method by Löhner et al., (1985b). It is a third-order diffusion term that is locally one-dimensional in the direction of the velocity vector. It uses the magnitude of the local velocity gradient as a sensor to locate a discontinuity. The second artificial dissipation method was developed by Peraire (Morgan and Peraire, 1987, and Peraire et al., 1988) for finite element computations. This method incorporates a sensor that is based upon the change in the local pressure gradient. Both artificial dissipation methods are used as filters. They are applied after a high-order solution has been obtained to control the nonphysical oscillations. Both methods have shown good results for flow simulations that do not contain strong discontinuities, such as normal shocks. In regions of strong discontinuities, monotonic behavior in the solution can generally not be maintained unless the diffusion coefficient is excessively large. This type of artificial dissipation may be preferred over high-resolution methods, such as FCT, for simulation of flows that do not contain strong discontinuities because they are far less costly from a computational point of view.

Nonlinear high-resolution monotone methods were invented to solve the problem of maintaining positivity and monotonicity with accuracy. The first nonlinear, monotone, positivity-preserving technique was the FCT algorithm developed by Boris and Book (1973-1976). The FCT technique constructs the net transportive flux point by point, nonlinearly, as a weighted average of a flux computed by a low-order scheme and a flux computed by a high-order scheme. The weighting is done in a manner which insures that the high-order flux is used to the greatest extent possible without introducing overshoots and undershoots. This weighting procedure is referred to as "flux-correction" or "flux limiting" because, in effect, the weighting is such that the fluxes are limited so that no new extrema in the solution are introduced. While Boris and Book's FCT solver is one-dimensional in nature, Zalesak (1979) developed a generalized, alternative approach to FCT that is truly multidimensional without introducing the flow direction bias of operator splitting. Zalesak's FCT algorithm allows the use of unstructured grids which makes the algorithm attractive for implementation into the finite element method. Löhner et al., (1987 and 1988) adapted Zalesak's FCT algorithm for

controlling the oscillations generated by the Two-Step Taylor-Galerkin FEM on unstructured meshes. This adaption, which resulted in the FEM-FCT solution scheme, is presented here as it is the high-resolution monotone method applied to the explicit predictor phase of the PCICE-FEM scheme.

2.4.1 Lapidus Artificial Diffusion

The Lapidus artificial dissipation method presented here strictly follows the form presented by Löhner et al., (1985b). As with many artificial dissipation schemes, Lapidus is a post-processing, or filtering, operation that is applied on the solution obtained from some monotonicity violating advection scheme (second-order or higher). For this study, the monotonicity violating scheme is the Two-Step Taylor-Galerkin FEM. Following Löhner's notation, the general form of the Lapidus artificial dissipation filtering operation is

$$U^{n+1} = U^h + \Delta t \frac{\partial}{\partial x_i} \left(k_v \frac{\partial U^h}{\partial x_i} \right), \quad (2.19)$$

where U^h is the high-order solution at time t^{n+1} that, in this case, is obtained from equation (2.17). The coefficient, k_v , is the artificial dissipation components in the x_i direction given by

$$k_v = C_k h_e^2 \left| \frac{\partial u_i}{\partial x_i} \right|,$$

where C_k is a user defined constant, u_i are the components of velocity in the i direction, and h_e is the element length. Equation (2.19) is applied on orthogonal finite difference grids in a one-dimensional operator split fashion. However, under coordinate rotation, equation (2.19) is not invariant. To avoid this problem, a local unit vector \vec{l} pointing in the direction of the maximum change in the absolute value of velocity is defined as

$$\vec{l} = \frac{\vec{\nabla} |\vec{u}|}{|\vec{\nabla} |\vec{u}||}. \quad (2.20)$$

Writing equation (2.19) in terms of the one-dimensional coordinate system l defined by the cartesian components of \vec{l} yields,

$$U^{n+1} = U^h + \Delta t \frac{\partial}{\partial l} \left(k_l \frac{\partial U^h}{\partial l} \right), \quad (2.21)$$

where k_l is the diffusion coefficient in terms of l ,

$$k_l = C_k h_e^2 \left| \frac{\partial(\vec{u} \cdot \vec{l})}{\partial l} \right|. \quad (2.22)$$

Equation (2.21) is now the locally one-dimensional form of Lapidus artificial diffusion in the direction of the maximum change in the magnitude of velocity. This direction is invariant under coordinate axis rotation which means that the stabilizing diffusion is grid orientation independent. k_l has the desired properties required of a variable artificial dissipation method for simulating compressible flows. Near shocks, where velocity is modeled as a sudden change (simulated discontinuity) over a few computational cells, the value of k_l is high. Near a contact discontinuity where the change in velocity normal to this surface is negligible, k_l vanishes and thus an unwanted cross-diffusion effect is avoided. k_l is also small in viscous boundary layers where the maximum change in velocity is normal to the velocity direction. Therefore, $\vec{u} \cdot \vec{l} \approx 0$ and, thus, $k_l \approx 0$. The result is that, in the boundary layer, only the physical viscosity effects are taken into account and not artificial dissipation.

Finite element discretization of Lapidus artificial diffusion centers around determining the derivatives in the direction of \vec{l} . This is a straightforward process with linear triangular and tetrahedral finite elements where the shape function derivatives are elemental constants (see Appendix A). With constant derivatives, \vec{l} and, thus, k_l will be determined on an elemental basis. By noting that

$$\frac{\vec{\nabla}(\vec{u}^2)}{|\vec{\nabla}(\vec{u}^2)|} = \frac{\vec{\nabla}|\vec{u}|}{|\vec{\nabla}|\vec{u}||}, \quad (2.23)$$

use \vec{u}^2 instead of $|\vec{u}|$ in equation (2.20) to compute \vec{l} . Employing finite element approximation theory, the elemental gradient of \vec{u}^2 can be expressed as

$$\vec{\nabla}(\vec{u}^2)_e = [\vec{\nabla}N]_e \{\vec{u}^2\}_e, \quad (2.24)$$

where $[\vec{\nabla}N]_e$ and $\{\vec{u}^2\}_e$ are the element's nodal shape function derivatives and nodal values of \vec{u}^2 , respectively. Once the components of \vec{l} have been determined, the projection of the shape function derivatives in the l direction can then be found,

$$\left\{ \frac{\partial N}{\partial l} \right\}_e = l_i \left\{ \frac{\partial N}{\partial x_i} \right\}_e. \quad (2.25)$$

The finite element discretization of Lapidus artificial diffusion, equation (2.21), can be accomplished by the standard Galerkin weighted residual method with substitution of equations (2.24) and (2.25),

$$\sum_e \int_{\Omega} \{N\} [N] d\Omega \{U^{n+1} - U^h\} = -\Delta t \sum_e \int_{\Omega} \left\{ \frac{\partial N}{\partial l} \right\} k_l \left[\frac{\partial N}{\partial l} \right] d\Omega \{U^h\}. \quad (2.26)$$

Green's theorem has been applied to the right-hand side of equation (2.26). Note, however, that there is no boundary integral in equation (2.26). In a personal communication with Löhner (1994), he stated that the boundary integral was neglected as exact evaluation of diffusion on the domain boundary was unnecessary. This was found to be true for the two-step Taylor-Galerkin FEM where Lapidus artificial diffusion performed quite well. Furthermore, the boundary integral would have presented some interesting problems with the boundary normal \vec{n} . Evaluation of a boundary integral containing \vec{n} where the derivatives are locally one-dimensional in the l direction would be difficult.

Application of equation (2.26) involves lumping the consistent mass matrix. After integration of equation (2.26), the new-time solution becomes

$$U^{n+1} = U^h - \Delta t M_l^{-1} \sum_e \Omega_e \left\{ \frac{\partial N}{\partial l} \right\} k_l \left[\frac{\partial N}{\partial l} \right] \{U^h\}. \quad (2.27)$$

2.4.2 Peraire's Artificial Dissipation

Peraire's artificial dissipation method (Morgan and Peraire, 1987 and Peraire et al., 1988) was developed for application in the finite element computations of compressible flow. It is derived from a modified form developed by MacCormack and Baldwin (1975) where the diffusion coefficient is based upon the second derivative of pressure. Peraire's main modification approximates the second derivatives with the difference between the consistent and mass-lumped matrices.

The original form of MacCormack's artificial viscosity, in a finite element context (Morgan and Peraire, 1987), is written as

$$U^{n+1} = U^h + \Delta t C_P h_e^3 \frac{\partial}{\partial x_i} \left(S_P \frac{\partial U^h}{\partial x_i} \right) \quad (2.28)$$

where

$$S_P = \frac{|\vec{u}| + c}{\bar{P}} \frac{\partial^2 P}{\partial x_j \partial x_j} \quad (2.29)$$

is the pressure sensor acting as an artificial dissipation coefficient. As before, U^h is the high-order solution at time t^{n+1} and h_e is the element length. P is the pressure field, \bar{P} is the average pressure over an element, and C_P is a user defined constant. The main reason for the factor

$$\frac{|\vec{u}| + c}{\bar{P}}$$

is to render equation (2.28) dimensionally correct. Notice that the right-hand side of equation (2.28) will require the evaluation of two second-order differentials by performing a computationally expensive variational recovery. To avoid this costly process, a diffusion operator can be approximated very cheaply by noting that the difference between the consistent mass and mass-lumped matrices approximates a diffusion effect. This diffusion effect, based upon the pressure field, can be written as

$$\sum_e \frac{\partial}{\partial x_i} \left(h_e^2 \frac{\partial P}{\partial x_i} \right) \equiv M_l^{-1} (M_c - M_l) \{P\}. \quad (2.30)$$

The right-hand side of equation (2.30) gives in one dimension an identical stencil to that obtained by applying the standard Galerkin weighted residual method to the diffusion term on the left-hand side of equation (2.30). For the multi-dimensional case, the exact equivalence is no longer valid but the artificial dissipation effects are still maintained. In any event, there is no need for exact diffusion because any added artificial dissipation is non-physical. The only requirements placed on artificial dissipation are that it stabilize the solution in the vicinity of shocks and that the solution remains conservative. Applying the concept embodied by equation (2.30) to equations (2.28) and (2.29), equation (2.28) can be replaced by

$$U^{n+1} = U^h + C_P M_l^{-1} \left[\frac{M_l^{-1} (M_c - M_l) \{P\}}{\bar{P}} (M_c - M_l) \{U^h\} \right]. \quad (2.31)$$

When used for complex flows, where several shocks with different strengths occur, normalization of the pressure sensor is required. This modification replaces \bar{P} with the sum of the first differences of P . This yields the final form of Peraire's artificial dissipation,

$$U^{n+1} = U^h + C_P M_l^{-1} \left[\frac{(M_c - M_l) \{P\}}{|(M_c - M_l) \{P\}|} (M_c - M_l) \{U^h\} \right]. \quad (2.32)$$

Equation (2.32) strictly follows Peraire's notation (Morgan and Peraire, 1987). However, this notation is somewhat ambiguous. Actual application of the pressure sensor on a linear triangular finite element is

$$\frac{(M_c - M_l)\{P\}}{|(M_c - M_l)\{P\}|} = \frac{\sum_e |2P_I - P_J - P_K|}{\sum_e [|P_I - P_J| + |P_J - P_K| + \sigma]}, \quad (2.33)$$

where I , J , and K refer to the node numbers of the element and σ is some small number to avoid division by zero in constant pressure regions. After assembling equation (2.33), the elemental artificial dissipation coefficient for this method is found by taking the minimum nodal value of the element. It always varies between 0 and 1 regardless of shock strength. It is a maximum in the shock and a minimum in smooth regions of flow or across contact discontinuities and slip lines.

2.4.3 Flux-Corrected Transport

The Flux-Corrected Transport (FCT) method presented here is Löhner's adaptation (Löhner et al., 1987 and 1988) of Zalesak's (1979) multi-dimensional FCT scheme. The main concept behind FCT is to combine a high-order scheme with a low-order scheme such that in regions where the flow variables vary smoothly the high-order scheme dominates, whereas in the regions where the flow variables vary rapidly and suffer from non-physical oscillations the two schemes are combined, in a conservative manner, to achieve near-monotonic behavior. The main requirement of the low-order scheme is that it be monotonic. Löhner's main contribution is in applying this scheme for systems of hyperbolic equations on unstructured finite elements. With no modification, the following FCT method of Löhner can also be applied to the explicit predictor phase of the PCICE-FEM scheme.

2.4.3.1 Flux-Corrected Transport for the Finite Element Method

Following Zalesak's development and keeping with Löhner's approach and notation, FCT for the finite element method consists of the following six algorithmic steps applied separately to each equation of the system of equations (1.2):

- 1) Compute LEC_{el} : the "low-order element contributions" from some low-order scheme guaranteed to give monotonic results for the problem at hand.
- 2) Compute HEC_{el} : the "high-order element contribution" from some high-order scheme.

- 3) Define AEC_{el} : the "antidiffusive element contributions":

$$AEC_{el} = HEC_{el} - LEC_{el} . \quad (2.34)$$

Löhner has indicated (personal communication, 1992) that the performance of FCT in regions near shocks can be improved by modifying the antidiffusive element contributions AEC_{el} . The sign on AEC_{el} is reversed if

$$\vec{\nabla} AEC \cdot \vec{\nabla} U^l < 0 .$$

- 4) Compute the updated low-order solution:

$$U_I^l = U_I^n + \sum_{el} LEC_{el} = U_I^n + \delta U_I^l , \quad (2.35)$$

where el implies the summation of all elemental contributions from the elements surrounding node I .

- 5) Limit or "correct" the AEC so that U^{n+1} as computed in Step 6 below is free of extrema not also found in U^l or U^n :

$$AEC_{el}^c = C_{el} \cdot AEC_{el} , 0 \leq C_{el} \leq 1 . \quad (2.36)$$

- 6) Apply the limited AEC :

$$U_I^{n+1} = U_I^l + \sum_{el} AEC_{el}^c . \quad (2.37)$$

2.4.3.2 The Low-Order Scheme: Lumped-Mass Taylor-Galerkin Plus Diffusion

As discussed earlier, the low-order scheme in any FCT method must be monotonic. The low-order scheme must therefore not produce artificial overshoots and undershoots. Generally speaking, the better the low-order scheme is, the "easier" the task of limiting. Mass-diffusion added to the lumped-mass Taylor-Galerkin scheme will yield a monotonic scheme. Following Löhner et al. (1987), the diffusion is obtained by subtracting the lumped-mass matrix from the consistent-mass matrix for linear (the only type considered here) elements. Hence, the diffusion over an element is defined as

$$DIFF = C_d (M_l - M_c) \{U\}^n \quad (2.38)$$

where C_d is the diffusion coefficient (usually $C_d = I$). The lumped-mass Taylor-Galerkin plus diffusion scheme is then defined as

$$M_l \delta U^l = R + DIFF \quad (2.39)$$

where δU^l is the incremental change in the low-order solution.

2.4.3.3 Antidiffusive Element Contributions

As defined above, the antidiffusive element contributions are the difference between the high-order solution and the low-order solution. Subtracting equation (2.39) from equation (2.17), yields

$$M_l(\delta U^h - \delta U^l) = (M_l - M_c)(U^n + \delta U^h). \quad (2.40)$$

Notice that all the terms arising from the discretization of the advective fluxes \vec{F} , which are contained in R , have now cancelled.

2.4.3.4 The FCT Limiting Procedure

The FCT procedure depends critically on the limiting step (Step 5), equation (2.36). Define the following quantities:

(a) P_I^+ and P_I^- : the sum of all positive (negative) antidiffusive element contributions to node I :

$$P_I^+ = \sum_{el} \max(0, AEC_{el}),$$

$$P_I^- = \sum_{el} \min(0, AEC_{el}).$$

(b) Q_I^+ and Q_I^- : the maximum increment and minimum decrement node I is allowed to achieve in equation (2.37):

$$Q_I^+ = U_I^{\max} - U^l,$$

$$Q_I^- = U_I^{\min} - U^l.$$

U_I^{\max} and U_I^{\min} (defined below) represent the maximum and minimum value, respectively, the unknown U at node I is allowed to achieve in (2.37).

(c) R_I^+ and R_I^- :

$$R_I^+ := \begin{cases} \min(1, Q_I^+ / P_I^+), & \text{if } P_I^+ > 0, \\ 0, & \text{if } P_I^+ = 0, \end{cases}$$

$$R_I^- := \begin{cases} \min(1, Q_I^- / P_I^-), & \text{if } P_I^- > 0, \\ 0, & \text{if } P_I^- = 0. \end{cases}$$

Now determine the limiter for each element:

$$C_{el} := \min(\text{nodal values}) \begin{cases} R_I^+, & \text{if } AEC > 0, \\ R_I^-, & \text{if } AEC < 0. \end{cases}$$

U_I^{\max} and U_I^{\min} , used above, are obtained in three steps:

(a) Maximum (minimum) nodal value of U_I^n and U_I^l :

$$\begin{aligned} U_I^{*\max} &= \max(U_I^l, U_I^n), \\ U_I^{*\min} &= \min(U_I^l, U_I^n). \end{aligned}$$

(b) Maximum (minimum) nodal value of an element is defined as:

$$\begin{aligned} U_{el}^{*\max} &= \max(U_A^{*\max}, U_B^{*\max}, \dots, U_C^{*\max}), \\ U_{el}^{*\min} &= \min(U_A^{*\min}, U_B^{*\min}, \dots, U_C^{*\min}), \end{aligned}$$

where A, B, \dots, C represents the nodes of element el .

(c) Maximum (minimum) U of all elements surrounding node I is denoted:

$$\begin{aligned} U_I^{\max} &= (U_1^{*\max}, U_2^{*\max}, \dots, U_m^{*\max}), \\ U_I^{\min} &= (U_1^{*\min}, U_2^{*\min}, \dots, U_m^{*\min}), \end{aligned}$$

where $1, 2, \dots, m$ represents the elements containing node I .

2.4.3.5 FCT Limiting for Systems of Equations

FCT produces perfectly monotonic results for a single scalar advection equation. However, when applying the limiting process to systems of equations there is no satisfactory way to limit each equation to maintain perfect monotonic behavior in the conserved variables. For systems of equations, the possible choices for the limiter C_{el} in equation (2.36) are many. The use of the same limiter C_{el} for all equations seems to produce the best results because the phase errors for all equations are synchronized (Löhner et al., 1987). The objective then is to find the limiter that best addresses the character of the governing equations. For the governing equation system, equation (1.2), the limiter that seems to produce the best results for step 5) of the limiting process, equation (2.36), seems to be the minimum of the limiters obtained from the density and total energy,

$$C_{el} = \min(C_{el}(\rho), C_{el}(\rho e_t)). \quad (2.41)$$

The limiter obtained from equation (2.41) will not produce strict monotonic results for strong shocks where undershoots of less than 1% for the conserved variables occur. However, this is much better performance than can be achieved from the use of artificial dissipation methods, such as Lapidus and Peraire.

2.5 Boundary Conditions

Boundary conditions are an integral part of the solution to partial differential equations. The governing equations of compressible flow, equations (1.2), are a system of fully coupled nonlinear first-order partial differential equations. This coupled system constitutes an initial-boundary value problem that requires both a set of initial conditions and boundary conditions in order to be solved. The initial conditions must satisfy the governing equations at some reference time level, which is generally at the beginning of the simulation. For finite domains, the boundary conditions supply the domain with information required to achieve a unique solution. In other words, a set of initial conditions and boundary conditions that satisfy the governing partial differential equations establish a well-posed mathematical problem. The topic of this section is the development and application of numerical boundary conditions for unstructured grids. The application of these boundary conditions are consistent with the order of accuracy and the stability criteria for the two-step Taylor-Galerkin FEM scheme.

All too often when a solution method is discussed in the literature, the topic of boundary conditions are glossed over or completely ignored. However, the proper treatment of boundary conditions for a particular scheme is as important as the scheme itself. For the hyperbolic wave propagation-dominated system of equations considered here, the three following questions (Hirsch, 1990) must be addressed:

- 1) How many specified variables must be imposed at a boundary for a given boundary condition type?
- 2) How are the remaining variables determined at the boundary?
- 3) How are the boundary conditions formulated and discretized in order to be consistent with the order of accuracy and stability criteria of the solution scheme for the interior of the domain?

The first two questions are answered by a characteristic analysis of the inviscid form of the governing equations. The derivation of the compatibility relations will follow the derivation contained in Chapters 16 and 19 of Hirsch (1990). It will be reproduced here to provide a clear understanding of characteristic theory for the system of hyperbolic first-order partial differential equations that govern inviscid compressible flow. The spatial and temporal discretization can then be shown for linear triangular finite elements. The application of characteristic boundary conditions is developed here for the two-step Taylor-Galerkin FEM, specifically for the convective flux boundary integral contained in equation (2.12). Characteristic boundary conditions will be provided for solid wall and symmetry boundaries, flow exits, and flow inlets. For the case of viscous compressible flow, a discussion of an appropriate treatment of the viscous flux boundary integral in equation (2.12) will end this chapter.

2.5.1 Compatibility Relations for Compressible Flow

The new application of numerical boundary conditions on unstructured grids presented here is based upon the wave-like solutions for the convective components of the governing equations. Each solution emits a wave that can be associated with a surface, called a characteristic surface, propagating in a characteristic direction with a characteristic velocity. The characteristic surface is mathematically defined by an appropriate linear combination of the governing equations, called compatibility equations, that contain only derivatives in the characteristic directions. The characteristic directions are found by rotating the flux vector Jacobian of the homogeneous part of the governing equations, equation (1.2), into the principle (characteristic) directions. The rotation transforms the governing equations into compatibility equations in terms of characteristic variables. When applied at the boundary of the domain, the characteristic directions define the information propagating into and out of the domain. Information propagating into the domain must be specified. Information propagating out of the domain must be obtained from the solution of the compatibility equation corresponding to the outgoing characteristic direction.

The following derivation of the compatibility equations will be taken from Hirsch (1990). In the context of this study, the derivation will be two-dimensional. For the determination of the convective flux boundary integral of equation (2.12), the solution of the compatibility relations applied at the boundary will provide the variables to construct the convective fluxes.

The derivation of the compatibility relations is based upon the quasi-linear form of the governing equations where only the non-diffusive terms in equation (1.2) are considered,

$$\frac{\partial U}{\partial t} + \left(\frac{\partial \vec{F}}{\partial U} \right) \cdot \vec{\nabla} U = Q, \quad (2.42)$$

or

$$\frac{\partial U}{\partial t} + \vec{A} \cdot \vec{\nabla} U = Q, \quad (2.43)$$

or in component form

$$\frac{\partial U}{\partial t} + A \frac{\partial U}{\partial x} + B \frac{\partial U}{\partial y} = Q, \quad (2.44)$$

where A and B are the Jacobian matrices of the convective flux vector \vec{F} . They can be condensed into vector form,

$$\vec{A} = \frac{\partial \vec{F}}{\partial U}, \quad (2.45)$$

having the components A and B defined by

$$A = \frac{\partial f}{\partial U} \quad B = \frac{\partial g}{\partial U}, \quad (2.46)$$

where f and g are the components of the convective flux vector \vec{F} defined in equation (1.1).

Determination of the Jacobian matrices can be simplified by writing the conserved variable vector U and the convective flux vector \vec{F} as functions of ρ , $m = \rho u$, $n = \rho v$, and $\varepsilon = \rho e_t$. With

$$U = \begin{Bmatrix} \rho \\ \rho u \\ \rho v \\ \rho e_t \end{Bmatrix} = \begin{Bmatrix} \rho \\ m \\ n \\ \varepsilon \end{Bmatrix}, \quad (2.47)$$

the flux vector components can be written as

$$f = \begin{Bmatrix} m \\ \frac{m^2}{\rho} + P \\ \frac{mn}{\rho} \\ \frac{m(\varepsilon + P)}{\rho} \end{Bmatrix} \quad g = \begin{Bmatrix} n \\ \frac{mn}{\rho} \\ \frac{n^2}{\rho} + P \\ \frac{n(\varepsilon + P)}{\rho} \end{Bmatrix}. \quad (2.48)$$

The Jacobian matrices can be determined explicitly if the equation of state for pressure P is specified. Rewriting equation (1.11) in terms of ρ , m , n , and ε yields

$$P = (\gamma - 1) \left(\rho e_t - \frac{\rho \vec{u} \cdot \vec{u}}{2} \right) = (\gamma - 1) \left(\varepsilon - \frac{m^2 + n^2}{2\rho} \right). \quad (2.49)$$

Substituting this expression into the flux vector components of equation (2.48) gives

$$f = \begin{Bmatrix} m \\ \frac{m^2}{\rho} + (\gamma - 1) \left(\varepsilon - \frac{m^2 + n^2}{2\rho} \right) \\ \frac{mn}{\rho} \\ \frac{m}{\rho} \left[\gamma \varepsilon - (\gamma - 1) \frac{m^2 + n^2}{2\rho} \right] \end{Bmatrix} \quad g = \begin{Bmatrix} n \\ \frac{mn}{\rho} \\ \frac{n^2}{\rho} + (\gamma - 1) \left(\varepsilon - \frac{m^2 + n^2}{2\rho} \right) \\ \frac{n}{\rho} \left[\gamma \varepsilon - (\gamma - 1) \frac{m^2 + n^2}{2\rho} \right] \end{Bmatrix}. \quad (2.50)$$

The Jacobian matrices A and B are formed by assembling the column vectors obtained by differentiating the corresponding flux vector components with respect to the variables ρ , m , n , and ε . A is found by

$$A = \frac{\partial f}{\partial U} = \left[\frac{\partial f}{\partial \rho}, \frac{\partial f}{\partial m}, \frac{\partial f}{\partial n}, \frac{\partial f}{\partial \varepsilon} \right]. \quad (2.51)$$

Completing the derivatives in equation (2.51) yields,

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\frac{m^2}{\rho^2} + (\gamma-1)\frac{m^2+n^2}{2\rho^2} & \frac{2m}{\rho} - (\gamma-1)\frac{m}{\rho} & (1-\gamma)\frac{n}{\rho} & \gamma-1 \\ -\frac{mn}{\rho^2} & \frac{n}{\rho} & \frac{m}{\rho} & 0 \\ -\frac{m}{\rho^2} \left[\gamma\varepsilon - (\gamma-1)\frac{m^2+n^2}{\rho} \right] & \frac{\gamma\varepsilon}{\rho} - \frac{\gamma-1}{2\rho^2}(3m^2+n^2) & (1-\gamma)\frac{mn}{\rho^2} & \frac{\gamma m}{\rho} \end{bmatrix}. \quad (2.52)$$

Substituting the conserved variables for the variable expressions ρ , m , n , and ε yields,

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ \frac{\gamma-3}{2}u^2 + \frac{\gamma-1}{2}v^2 & (3-\gamma)u & (1-\gamma)v & \gamma-1 \\ -uv & v & u & 0 \\ -\gamma ue_t + (\gamma-1)u\vec{u}^2 & \gamma e_t - \frac{\gamma-1}{2}(3u^2+v^2) & (1-\gamma)uv & \gamma u \end{bmatrix}. \quad (2.53)$$

B is found by

$$B = \frac{\partial g}{\partial U} = \left[\frac{\partial g}{\partial \rho}, \frac{\partial g}{\partial m}, \frac{\partial g}{\partial n}, \frac{\partial g}{\partial \varepsilon} \right], \quad (2.54)$$

which results in

$$B = \begin{bmatrix} 0 & 0 & 1 & 0 \\ -uv & v & u & 0 \\ \frac{\gamma-1}{2}u^2 + \frac{\gamma-3}{2}v^2 & (1-\gamma)u & (\gamma-3)v & \gamma-1 \\ -\gamma ve_t + (\gamma-1)v\vec{u}^2 & (1-\gamma)uv & \gamma e_t - \frac{\gamma-1}{2}(u^2+3v^2) & \gamma v \end{bmatrix}. \quad (2.55)$$

The classification of equation (2.43) as hyperbolic can be determined by finding the eigenvalues of linear combinations of A and B . Equation (2.43) is hyperbolic if the matrix K , defined by

$$K = \vec{A} \cdot \vec{\kappa} = A\kappa_x + B\kappa_y, \quad (2.56)$$

has real eigenvalues for any set of values of $\vec{\kappa}$. Later, K will be applied at the boundary and the arbitrary unit direction vector $\vec{\kappa}$ will be replaced by the unit outward normal vector to the boundary $\vec{\eta}$. The eigenvalues of K are found by

$$\det|\lambda\bar{I} - K| = 0, \quad (2.57)$$

where λ is the set of eigenvalues and \bar{I} is the identity matrix. Hirsch (1990) gives the eigenvalues of this system as

$$\begin{aligned} \lambda_1 &= \lambda_2 = \vec{u} \cdot \vec{\kappa} \\ \lambda_3 &= \vec{u} \cdot \vec{\kappa} + c \\ \lambda_4 &= \vec{u} \cdot \vec{\kappa} - c. \end{aligned} \quad (2.58)$$

This solution of equation (2.57) was verified with the use of MATHEMATICA[®] (1996) after substituting the following thermodynamic relation for specific total energy into the flux Jacobians A and B ,

$$e_t = \frac{c^2}{\gamma(\gamma - 1)} + \frac{\vec{u}^2}{2}. \quad (2.59)$$

Equation (2.59) was derived by substituting

$$c^2 = \frac{\gamma P}{\rho}$$

into equation (1.11) for pressure P and solving for e_t .

Wave-like solutions will exist if the eigenvalues of the matrix $K = \vec{A} \cdot \vec{\kappa}$, for arbitrary $\vec{\kappa}$, are real with linear independence of the corresponding left eigenvectors. The eigenvalues given by equation (2.58) are real and linearly independent and thus verifying the hyperbolic nature of equation (2.43). Later, when discussing the general properties of characteristics, the eigenvalues given in equation (2.58) will be shown to be the characteristic directions. The set of left eigenvectors for K will be used as the rotation matrix to transform the governing equations (flow variable space) to the compatibility equations (characteristic wave propagation space). There is a problem, however, with the left eigenvectors of the conservative variable form of the matrix K . Manipulation of the matrix K to determine the left eigenvectors is too difficult by hand. MATHEMATICA[®] (1996) produced a set of left eigenvectors so complex as to make the derivation of the compatibility equations intractable. The solution then is to employ a simpler set of governing equations that have the same characteristic properties as those of equation (2.43).

The conserved variable system of equations, equation (2.43), can be transformed into non-conservative variable form by the transformation matrix

$$M = \frac{\partial U}{\partial V}, \quad (2.60)$$

where V is the vector of non-conservative, or primitive, variables,

$$V = \begin{Bmatrix} \rho \\ u \\ v \\ P \end{Bmatrix}.$$

Using the thermodynamic relation for pressure, equation (1.11), the transformation matrix M is defined by

$$M = \left[\frac{\partial U}{\partial \rho}, \frac{\partial U}{\partial u}, \frac{\partial U}{\partial v}, \frac{\partial U}{\partial P} \right] = \begin{bmatrix} 1 & 0 & 0 & 0 \\ u & \rho & 0 & 0 \\ v & 0 & \rho & 0 \\ \frac{\vec{u}^2}{2} & \rho u & \rho v & \frac{1}{\gamma-1} \end{bmatrix}, \quad (2.61)$$

and its inverse is

$$M^{-1} = \left[\frac{\partial V}{\partial \rho}, \frac{\partial V}{\partial m}, \frac{\partial V}{\partial n}, \frac{\partial V}{\partial \varepsilon} \right] = \begin{bmatrix} 1 & 0 & 0 & 0 \\ \frac{-u}{\rho} & \frac{1}{\rho} & 0 & 0 \\ \frac{-v}{\rho} & 0 & \frac{1}{\rho} & 0 \\ \frac{(\gamma-1)\vec{u}^2}{2} & (1-\gamma)u & (1-\gamma)v & \gamma-1 \end{bmatrix}. \quad (2.62)$$

The relationship between the conservative variables and non-conservative variable flux Jacobians can be expressed through the similarity transformation with matrix M . Introducing equation (2.60) into equation (2.43) leads to

$$M \frac{\partial V}{\partial t} + \vec{A} M \cdot \vec{\nabla} V = Q, \quad (2.63)$$

or after multiplication by M^{-1} ,

$$\frac{\partial V}{\partial t} + M^{-1} \vec{A} M \cdot \vec{\nabla} V = M^{-1} Q. \quad (2.64)$$

The non-conservative variable flux Jacobians can now be written as

$$\vec{\tilde{A}} = M^{-1} \vec{A} M, \quad (2.65)$$

which leads to the components of the non-conservative variable flux Jacobians,

$$\tilde{A} = M^{-1} A M = \begin{bmatrix} u & \rho & 0 & 0 \\ 0 & u & 0 & \frac{1}{\rho} \\ 0 & 0 & u & 0 \\ 0 & \rho c^2 & 0 & u \end{bmatrix} \quad (2.66)$$

and

$$\tilde{B} = M^{-1} B M = \begin{bmatrix} v & 0 & \rho & 0 \\ 0 & v & 0 & 0 \\ 0 & 0 & v & \frac{1}{\rho} \\ 0 & 0 & \rho c^2 & v \end{bmatrix}. \quad (2.67)$$

For the source terms,

$$\tilde{Q} = M^{-1} Q = \begin{Bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{Bmatrix}, \quad (2.68)$$

where q_1 , q_2 , q_3 , and q_4 are the sources for mass, the momentum components, and energy, respectively. Substituting relations (2.65) and (2.68) into equation (2.64) yields,

$$\frac{\partial V}{\partial t} + \vec{\tilde{A}} \cdot \vec{\nabla} V = \tilde{Q}, \quad (2.69)$$

or

$$\frac{\partial V}{\partial t} + \tilde{A} \frac{\partial V}{\partial x} + \tilde{B} \frac{\partial V}{\partial y} = \tilde{Q}. \quad (2.70)$$

In the absence of source terms, equation (2.69) defines the components of the primitive variable form of the governing equations as

$$\frac{\partial \rho}{\partial t} + \vec{u} \cdot \vec{\nabla} \rho + \rho \vec{\nabla} \cdot \vec{u} = 0, \quad (2.71)$$

$$\frac{\partial \vec{u}}{\partial t} + (\vec{u} \cdot \vec{\nabla}) \vec{u} + \frac{1}{\rho} \vec{\nabla} P = 0, \quad (2.72)$$

$$\frac{\partial P}{\partial t} + \vec{u} \cdot \vec{\nabla} P + \rho c^2 \vec{\nabla} \cdot \vec{u} = 0. \quad (2.73)$$

A primitive variable matrix, defined as

$$\tilde{K} = \tilde{A} \cdot \vec{K} = \tilde{A} \kappa_x + \tilde{B} \kappa_y = \begin{bmatrix} \vec{u} \cdot \vec{K} & \rho \kappa_x & \rho \kappa_y & 0 \\ 0 & \vec{u} \cdot \vec{K} & 0 & \frac{\kappa_x}{\rho} \\ 0 & 0 & \vec{u} \cdot \vec{K} & \frac{\kappa_y}{\rho} \\ 0 & \rho c^2 \kappa_x & \rho c^2 \kappa_y & \vec{u} \cdot \vec{K} \end{bmatrix}, \quad (2.74)$$

must have the same eigenvalues as K , given in equation (2.58), in order for the primitive variable system, equation (2.69), to represent the characteristic properties of the conservative variable system, equation (2.43). As with the conservative variable system, wave-like solutions for the primitive variable system will exist if the eigenvalues of equation (2.74) are real with linear independence of the corresponding left eigenvectors. In other words, if $\lambda_{(j)}$ denotes the j th eigenvalue of the matrix \tilde{K} , obtained from

$$\det |\lambda \bar{I} - \tilde{K}| = 0, \quad (2.75)$$

the left eigenvectors $l^{(j)}$ are solutions of

$$l^{(j)} \tilde{K} = \lambda_{(j)} l^{(j)}. \quad (2.76)$$

For each eigenvalue $\lambda_{(j)}$ and unit direction vector $\vec{\kappa}$, a wave phase surface, or characteristic surface, defined by

$$S(\vec{x}, t) = \vec{\kappa} \cdot \vec{x} - \omega t = C, \quad (2.77)$$

normal to vector $\vec{\kappa}$ at an instant in time has the characteristic properties in space and time of

$$\frac{\partial S}{\partial t} = -\lambda_{(j)} \quad (2.78)$$

and

$$\vec{\nabla} S = \vec{\kappa}. \quad (2.79)$$

In equation (2.77), ω is the frequency and C is a constant. By inspection of equation (2.78), ω in equation (2.77) is defined as

$$\omega = +\lambda_{(j)}. \quad (2.80)$$

Therefore, the eigenvalues of \tilde{K} represent the frequency ω , up to a factor of 2π , of the propagating wave. This wave propagates with a phase velocity, or characteristic velocity, \vec{a} , defined by

$$\vec{a} = +\lambda_{(j)} \vec{\kappa} \quad (2.81)$$

in the direction $\vec{\kappa}$, that is normal to the constant characteristic surface. The characteristic surface will follow this direction because

$$\frac{dS}{dt} = \frac{\partial S}{\partial t} + \vec{a} \cdot \vec{\nabla} S = -\lambda + \vec{a} \cdot \vec{\kappa} = 0 \quad (2.82)$$

along

$$\frac{d\vec{x}}{dt} = \vec{a} = \lambda \vec{\kappa}. \quad (2.83)$$

From equation 2.76, a system of left eigenvectors can be found that will diagonalize the matrix \tilde{K} . Constructing a matrix L^{-1} with the left eigenvectors $l^{(j)}$, the j th line of L^{-1} which is

the left eigenvector $l^{(j)}$ corresponding to the j th eigenvalue $\lambda_{(j)}$, equation (2.76) for all of the eigenvalues can be written as

$$L^{-1}\tilde{K} = \Lambda L, \quad (2.84)$$

where Λ is the diagonal matrix of the eigenvalues. With Λ in the direction of \vec{k} ,

$$\tilde{K} = L\Lambda L^{-1} \quad (2.85)$$

or

$$\Lambda = L^{-1}(\vec{A} \cdot \vec{k})L. \quad (2.86)$$

An important point to be made here is that the left eigenvectors can diagonalize any linear combination of \tilde{A} and \tilde{B} , such as $\vec{A} \cdot \vec{k}$ in equation (2.86). However, it is not possible to simultaneously diagonalize \tilde{A} and \tilde{B} because \tilde{A} and \tilde{B} do not have the same eigenvalues and eigenvectors. In other words, both \tilde{A} and \tilde{B} cannot be diagonalized by the same matrix, only a linear combination of the two.

With the introduction of the matrices L^{-1} and L , the compatibility equations can now be determined. Multiplying equation (2.69) by L^{-1} yields

$$L^{-1} \frac{\partial V}{\partial t} + L^{-1} \vec{A} \cdot \vec{\nabla} V = L^{-1} \tilde{Q}, \quad (2.87)$$

which is the set of compatibility equations for the primitive variable system of equations. Focusing on the first term in equation (2.87), L^{-1} acts as a transformation matrix between the primitive variables and a new set of variables, termed characteristic variables, defined by

$$\delta W = L^{-1} \delta V, \quad (2.88)$$

where δ refers to incremental variations in either time or space. Employing the identity relation $L^{-1}L = 1$, equation (2.87) can be written as

$$L^{-1} \frac{\partial V}{\partial t} + (L^{-1} \vec{A} L) \cdot L^{-1} \vec{\nabla} V = L^{-1} \tilde{Q}, \quad (2.89)$$

or in terms of the characteristic variables

$$\frac{\partial W}{\partial t} + (L^{-1} \vec{A} L) \cdot \vec{\nabla} W = L^{-1} \tilde{Q}. \quad (2.90)$$

The definition of characteristic variables in equation (2.88) expresses the spatial or temporal characteristic variable increments δW as a linear combination of the spatial or temporal primitive variable increments δV with coefficients of the components of the left eigenvectors. Because the left eigenvectors are functions of the flow variables, the coefficients are not constant. The integrability conditions cannot be satisfied and the characteristic variables W cannot be defined. However, δW always exists. This distinction will become apparent in the next section when discussing the solution of the compatibility equations.

To determine the left eigenvectors, write equation (2.76) for each eigenvalue. Writing this equation explicitly for a given eigenvalue $\lambda_{(j)}$, the components l_1, l_2, l_3 , and l_4 of the left eigenvector $l^{(j)}$ are the solutions of

$$[l_1 \quad l_2 \quad l_3 \quad l_4] \begin{bmatrix} u_\kappa & \rho \kappa_x & \rho \kappa_y & 0 \\ 0 & u_\kappa & 0 & \frac{\kappa_x}{\rho} \\ 0 & 0 & u_\kappa & \frac{\kappa_y}{\rho} \\ 0 & \rho c^2 \kappa_x & \rho c^2 \kappa_y & u_\kappa \end{bmatrix} = \lambda_{(j)} [l_1 \quad l_2 \quad l_3 \quad l_4], \quad (2.91)$$

where $u_\kappa = \vec{u} \cdot \vec{\kappa}$. For the first eigenvalue $\lambda_1 = u_\kappa$, equation (2.91) becomes

$$\begin{aligned} l_1 u_\kappa &= l_1 u_\kappa \\ l_1 \rho \kappa_x + l_2 u_\kappa + l_4 \rho c^2 \kappa_x &= l_2 u_\kappa \\ l_1 \rho \kappa_y + l_3 u_\kappa + l_4 \rho c^2 \kappa_y &= l_3 u_\kappa \\ \frac{l_2 \kappa_x}{\rho} + \frac{l_3 \kappa_y}{\rho} + l_4 u_\kappa &= l_4 u_\kappa. \end{aligned} \quad (2.92)$$

This equation system for the components of the first eigenvector contains four unknowns, l_1, l_2, l_3 , and l_4 , but only three linearly independent equations (note $l_1 = l_1$ in the first equation). This is due to the repeated eigenvalues of $\lambda_1 = \lambda_2 = u_\kappa$. Therefore, one of the eigenvector components can be arbitrarily chosen. Because the components of this first eigenvector make up the first row of L^{-1} , choose $l_1 = 1$. This will place a 1 on the diagonal of L^{-1} . The remaining

components are then easily found, $l_2 = l_3 = 0$ and $l_4 = -1/c^2$. The first eigenvector can then be written as

$$l^1 = \begin{bmatrix} 1 & 0 & 0 & -\frac{1}{c^2} \end{bmatrix}.$$

For the second eigenvalue $\lambda_2 = u_\kappa$, the same equation system (2.92) is used to find l^2 . In this instance, choose $l_1 = 0$. This results in

$$l^2 = \begin{bmatrix} 0 & \kappa_x & -\kappa_y & 0 \end{bmatrix}.$$

For the third and fourth eigenvalues, $\lambda_3 = u_\kappa + c$ and $\lambda_4 = u_\kappa - c$, the eigenvector component system becomes

$$\begin{aligned} l_1 u_\kappa &= l_1 (u_\kappa \pm c) \\ l_1 \rho \kappa_x + l_2 u_\kappa + l_4 \rho c^2 \kappa_x &= l_2 (u_\kappa \pm c) \\ l_1 \rho \kappa_y + l_3 u_\kappa + l_4 \rho c^2 \kappa_y &= l_3 (u_\kappa \pm c) \\ \frac{l_2 \kappa_x}{\rho} + \frac{l_3 \kappa_y}{\rho} + l_4 u_\kappa &= l_4 (u_\kappa \pm c), \end{aligned} \tag{2.93}$$

which results in

$$l^3 = \begin{bmatrix} 0 & \kappa_x & \kappa_y & \frac{1}{\rho c} \end{bmatrix}$$

and

$$l^4 = \begin{bmatrix} 0 & -\kappa_x & -\kappa_y & \frac{1}{\rho c} \end{bmatrix}.$$

The eigenvector matrices L^{-1} and L , or diagonalization matrices, take the form of

$$L^{-1} = \begin{bmatrix} 1 & 0 & 0 & -\frac{1}{c^2} \\ 0 & \kappa_y & -\kappa_x & 0 \\ 0 & \kappa_x & \kappa_y & \frac{1}{\rho c} \\ 0 & -\kappa_x & -\kappa_y & \frac{1}{\rho c} \end{bmatrix} \tag{2.94}$$

and

$$L = \begin{bmatrix} 1 & 0 & \frac{\rho}{2c} & \frac{\rho}{2c} \\ 0 & \kappa_y & \frac{\kappa_x}{2} & -\frac{\kappa_x}{2} \\ 0 & -\kappa_x & \frac{\kappa_y}{2} & -\frac{\kappa_y}{2} \\ 0 & 0 & \frac{\rho c}{2} & \frac{\rho c}{2} \end{bmatrix}. \quad (2.95)$$

The characteristic variables can now be determined from equation (2.88),

$$\delta W = \begin{Bmatrix} \delta w_1 \\ \delta w_2 \\ \delta w_3 \\ \delta w_4 \end{Bmatrix} = \begin{Bmatrix} \delta \rho - \frac{\delta P}{c^2} \\ \delta u \kappa_y - \delta v \kappa_x \\ \delta u \kappa_x + \delta v \kappa_y + \frac{\delta P}{\rho c} \\ -\delta u \kappa_x - \delta v \kappa_y + \frac{\delta P}{\rho c} \end{Bmatrix}, \quad (2.96)$$

where the linear combination of flow variables becomes apparent.

The compatibility, or characteristic, equations can now be obtained by writing the space gradient term of equation (2.90) in terms of a new vector as

$$\frac{\partial W}{\partial t} + \vec{Z} \cdot L \vec{\nabla} W = L^{-1} \tilde{Q}, \quad (2.97)$$

where

$$\vec{Z} = L^{-1} \vec{A}.$$

Equation (2.97) can be expanded into the system of compatibility equations for the governing primitive variable system of equations,

$$\frac{\partial w_1}{\partial t} + \vec{u} \cdot \vec{\nabla} w_1 = q_1 - \frac{q_2}{c^2}, \quad (2.98)$$

$$\frac{\partial w_2}{\partial t} + \vec{u} \cdot \vec{\nabla} w_2 = \frac{c}{2} \left[\kappa_x \frac{\partial(w_3 + w_4)}{\partial y} - \kappa_y \frac{\partial(w_3 + w_4)}{\partial x} \right] + q_2 \kappa_y - q_3 \kappa_x, \quad (2.99)$$

$$\frac{\partial w_3}{\partial t} + (\vec{u} + c\vec{\kappa}) \cdot \vec{\nabla} w_3 = c \left(\kappa_x \frac{\partial w_2}{\partial y} - \kappa_y \frac{\partial w_2}{\partial x} \right) + q_2 \kappa_x + q_3 \kappa_y + \frac{q_4}{\rho c}, \quad (2.100)$$

$$\frac{\partial w_4}{\partial t} + (\vec{u} - c\vec{\kappa}) \cdot \vec{\nabla} w_4 = c \left(\kappa_x \frac{\partial w_2}{\partial y} - \kappa_y \frac{\partial w_2}{\partial x} \right) - q_2 \kappa_x - q_3 \kappa_y + \frac{q_4}{\rho c}. \quad (2.101)$$

Note that the characteristic variables w_1 , w_2 , w_3 , and w_4 in equations (2.98)-(2.101) are never explicitly defined, only the increments in time and space are defined. In the next section, the time and space discretizations of equations (2.98)-(2.101) at the boundary, and how to recover the flow variables at the boundary, will be discussed.

2.5.2 Spatial and Temporal Discretization

As discussed previously, the characteristic equations, equations (2.98)-(2.101), can be written for an arbitrary propagation direction. In this section, these equations will be applied at the boundary of a discretized domain in the direction of the unit outward normal vector to the boundary. The first topic will be which of the characteristic equations must be applied at the boundary and thus what information must be specified at the boundary for a given boundary condition type. The incremental change in the characteristic variables, the corresponding characteristic directions, and the characteristic equations governing the propagation of the characteristic variables in terms of the unit outward normal vector to the boundary will then be given. And finally, the characteristic equations will be discretized in a fashion that is consistent with the spatial and temporal discretizations of the two-step Taylor-Galerkin FEM scheme.

The number of physical variables that can be specified at a boundary is dependent on the wave propagation properties of the system. The information that can be specified for a given boundary condition type is determined by the characteristic waves propagating from the boundary into the interior of the domain. The information that is specified by the boundary condition type is commonly referred to as physical boundary conditions. The remaining flow variables defined at the boundary must then be determined by the characteristic waves

propagating from the interior of the domain to the boundary. These wave solutions are defined by the characteristic equations, equations (2.98)-(2.101). The flow variables at the boundary that are determined by the characteristic equations are termed numerical boundary conditions. Over and under-specification of either the physical or numerical boundary conditions will result in an inconsistent, or ill-posed, mathematical formulation and will generally lead to a numerically unstable simulation. The boundary conditions must be consistent with the physical properties of the flow for a given boundary condition type and must as well be compatible with the discretized governing equations.

The incremental change in the characteristic variables in terms of the unit outward normal vector to the boundary $\vec{\eta}$ is

$$\delta W = \begin{Bmatrix} \delta w_1 \\ \delta w_2 \\ \delta w_3 \\ \delta w_4 \end{Bmatrix} = \begin{Bmatrix} \delta \rho - \frac{\delta P}{c^2} \\ -\delta u_\tau \\ \delta u_\eta + \frac{\delta P}{\rho c} \\ -\delta u_\eta + \frac{\delta P}{\rho c} \end{Bmatrix}, \quad (2.102)$$

where u_η is the normal velocity to the boundary defined by

$$u_\eta = \vec{u} \cdot \vec{\eta} = u\eta_x + v\eta_y. \quad (2.103)$$

The components η_x and η_y are the cartesian components of $\vec{\eta}$. The second incremental characteristic variable in equation (2.102), δw_2 , is in terms of the change in the tangential velocity at the boundary where the tangential velocity is defined by

$$u_\tau = \vec{u} \cdot \vec{\tau} = u\tau_x + v\tau_y. \quad (2.104)$$

The components τ_x and τ_y of the tangential vector to the boundary $\vec{\tau}$ are related, by the right-hand rule of vectors, to the components of $\vec{\eta}$ by $\tau_x = -\eta_y$ and $\tau_y = \eta_x$. Thus, the tangential component of velocity can be rewritten as

$$u_\tau = -u\eta_y + v\eta_x. \quad (2.105)$$

The characteristic directions at the boundary are defined by the eigenvalues of $\tilde{K} = \tilde{A} \cdot \vec{\eta}$,

$$\begin{aligned}\lambda_1 &= \lambda_2 = u_\eta, \\ \lambda_3 &= u_\eta + c, \\ \lambda_4 &= u_\eta - c.\end{aligned}\tag{2.106}$$

Along these characteristic directions, the variations in the characteristic variables are governed by equations (2.98)-(2.101) in terms of $\vec{\eta}$,

$$\frac{\partial w_1}{\partial t} + \vec{u} \cdot \vec{\nabla} w_1 = q_1 - \frac{q_2}{c^2},\tag{2.107}$$

$$\frac{\partial w_2}{\partial t} + \vec{u} \cdot \vec{\nabla} w_2 = \frac{c}{2} \vec{\tau} \cdot \vec{\nabla} (w_3 + w_4) + q_2 \eta_y - q_3 \eta_x,\tag{2.108}$$

$$\frac{\partial w_3}{\partial t} + (\vec{u} + c\vec{\eta}) \cdot \vec{\nabla} w_3 = c\vec{\tau} \cdot \vec{\nabla} w_2 + q_2 \eta_x + q_3 \eta_y + \frac{q_4}{\rho c},\tag{2.109}$$

$$\frac{\partial w_4}{\partial t} + (\vec{u} - c\vec{\eta}) \cdot \vec{\nabla} w_4 = c\vec{\tau} \cdot \vec{\nabla} w_2 - q_2 \eta_x - q_3 \eta_y + \frac{q_4}{\rho c}.\tag{2.110}$$

Equation (2.107) describes the constancy of the entropy along a streamline. Equation (2.108) defines the change in tangential velocity at the boundary and represents the propagation of vorticity waves. Equations (2.109) and (2.110) are associated with the acoustic waves of celerity $u_\eta \pm c$.

The numerical solutions of the applicable characteristic equations for a given boundary condition type constitutes the numerical boundary conditions. Coupled with the physical boundary conditions, the primitive flow variables at the boundary can be recovered from the incremental change in characteristic variables found in equation (2.102). For this study, the primitive flow variables are then used to construct the conservative flux vector $\vec{F}_e^{n+\frac{1}{2}}$ in the convective flux boundary integral of equation (2.12). This conservative flux vector in the unit outward normal direction, for a boundary face of element e , is defined as

$$\vec{\eta} \cdot \vec{F}_e^{n+\frac{1}{2}} = \left\{ \begin{array}{c} \rho u_\eta \\ \rho u u_\eta + P \eta_x \\ \rho v u_\eta + P \eta_y \\ \rho u_\eta H \end{array} \right\}^{n+\frac{1}{2}}.\tag{2.111}$$

Before any discussion of discretizing the characteristic equations can take place, it must first be established which characteristic equations are (or are not) to be applied at the boundary for a given boundary condition type (set of physical boundary conditions). This is determined by the wave propagation properties of the system given by the eigenvalues of $\tilde{K} = \tilde{A} \cdot \vec{\eta}$ found in equation (2.106). The sign of the eigenvalues determine whether the waves are propagating into or out of the domain. At a boundary, only the characteristic equations corresponding to positive (outgoing) eigenvalues may be applied because they provide the information propagating from the interior of the domain. The characteristic equations with negative (incoming) eigenvalues have to be replaced by the physical boundary conditions for a given boundary condition type. For example, equation (2.107) holds along the characteristic path defined by the first eigenvalue of equation (2.106),

$$\frac{d\vec{x}}{dt} = \frac{d\vec{\eta}}{dt} = \lambda_1 = u_n.$$

If the normal velocity u_n has a positive sign, then the information for the first characteristic variable w_1 is propagating from the interior of the domain to the boundary. Therefore, equation (2.107) must be solved numerically to provide the information for this characteristic at the boundary. If u_n has a negative sign, the information for w_1 is being propagated from the boundary into the interior of the domain. For a negative u_n , information for δw_1 has to be specified or a relation for δw_1 in terms of other flow variables must be specified.

Multi-dimensional flows are subject to a variety of boundaries of different types. These boundaries can be grouped into three main types; free-slip solid walls and symmetry plane, flow exits (subsonic or supersonic), and flow inlets (subsonic or supersonic). Figure 2-1 depicts the characteristic paths in the space and time plane for a free-slip solid wall or symmetry boundary condition. Note that Figure 2-1 does not include viscous no-slip solid

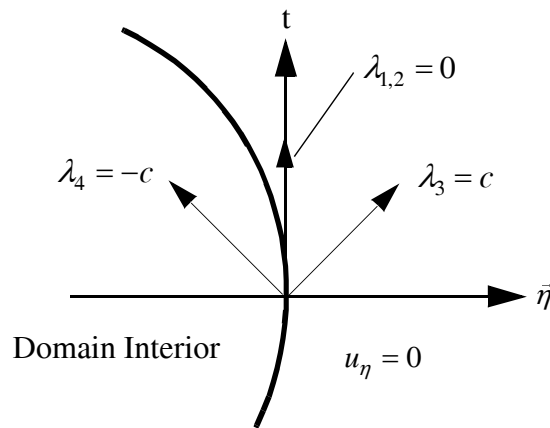


Figure 2-1 Characteristic directions at a free-slip solid wall or symmetry boundary.

walls where the characteristic formulation does not apply. The governing hydrodynamic

equations are purely elliptic at viscous solid walls. For an inviscid free-slip solid wall, there is one positive eigenvalue, λ_3 , and one negative eigenvalue, λ_4 . Therefore, only equation (2.109) must be solved numerically for δw_3 and δw_4 must be replaced with a physical boundary condition, which in this case is $u_\eta = 0$.

Figure 2-2 illustrates the characteristic directions in the space and time plane at an exit boundary for both subsonic and supersonic flows. In both cases the normal velocity u_η is

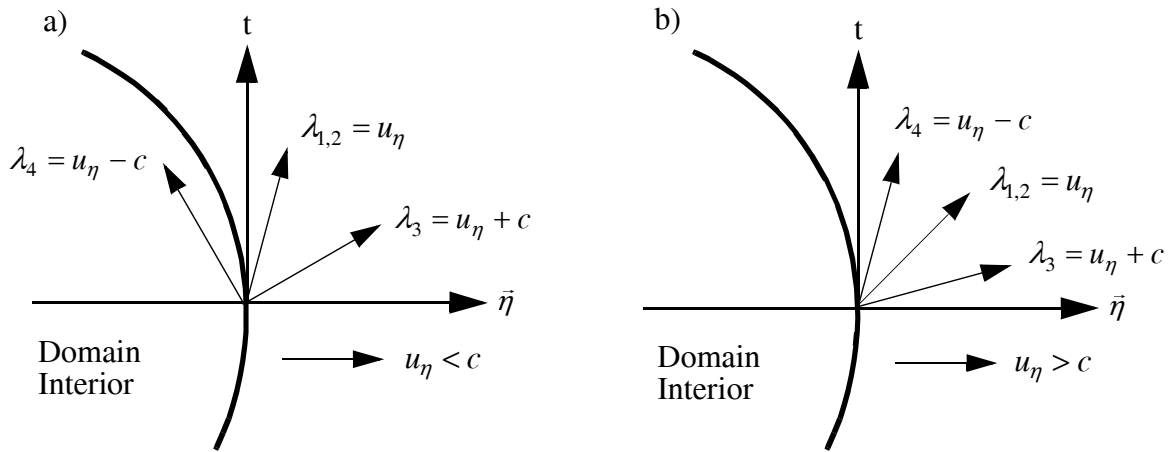


Figure 2-2 Characteristic directions for exit boundary conditions: a) subsonic b) supersonic.

always positive and, therefore, λ_1 , λ_2 , and λ_3 are always positive. Thus, the characteristic equations corresponding to these characteristic equations, equations (2.107) - (2.109) must be discretized at the boundary to obtain δw_1 , δw_2 , and δw_3 . In the subsonic exit case, $u_\eta < c$ results in λ_4 being negative. Therefore, δw_4 must be replaced with a physical boundary condition. For the supersonic case, $u_\eta > c$ and thus λ_4 is positive. This indicates, that for a supersonic exit, no physical boundary conditions may be specified. All of the characteristic equations, equations (2.107) - (2.110), must be numerically solved to obtain the boundary values of the flow variables.

Figure 2-3 the characteristic directions in the space and time plane at an inlet boundary for both subsonic and supersonic flows. At a flow inlet boundary, the normal velocity u_n is always negative and, therefore, λ_1 , λ_2 , and λ_4 are always negative. Thus, δw_1 , δw_2 , and δw_4 must be replaced by physical boundary conditions. In the subsonic inlet case, $u_\eta < c$ results in λ_3 being positive. Therefore, equation (2.109) must be solved numerically in order to obtain δw_3 . For the supersonic case, $u_\eta > c$ and all of the eigenvalues are negative. Thus, for a supersonic inlet, all of the boundary conditions must be specified.

Once it has been determined which of the characteristic equations must be applied at the boundary, the question of how these equations should be discretized for a global discretization scheme can be addressed. As discussed earlier, the characteristic equations must be discretized in a fashion that is consistent with the order of accuracy and stability of the global discretization scheme. This is a necessary requirement so that the global accuracy of the

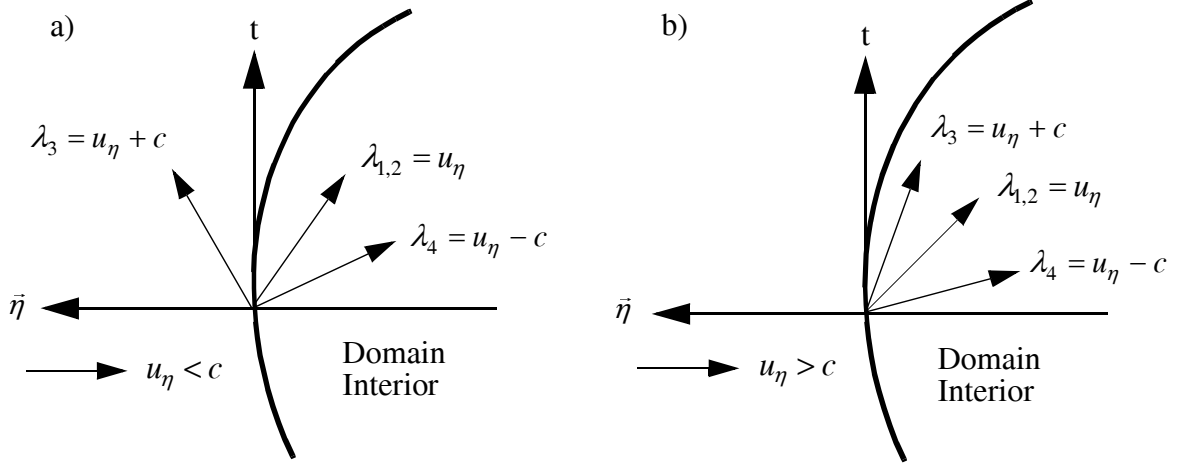


Figure 2-3 Characteristic directions for inlet boundary conditions: a) subsonic b) supersonic.

solution is not affected by the application of boundary conditions. Ideally, using the same discretization scheme for the boundary of the domain as the interior of the domain would satisfy this requirement. This is quite nearly possible in the case of the two-step Taylor-Galerkin FEM scheme. The application of the first step of this scheme, equation (2.3), can be applied to the characteristic equations, equations (2.107) - (2.110), with only slight variation.

Application of the first step of the two-step Taylor-Galerkin FEM scheme on the characteristic equations will provide the necessary values of the flow variables at time $t^{n+\frac{1}{2}}$ for the convective flux vector in the outward normal direction, equation (2.111), and for the convective flux boundary integral found in equation (2.12). There are several possible variations of temporal and spatial discretizations of equations (2.107) - (2.110) that would be consistent with the first step. However, the following discretization of the characteristic equations has proven to provide the best results. Using the forward-time temporal discretization found in equation (2.3), the temporal discretization of the characteristic equations from time t^n to $t^{n+\frac{1}{2}}$ is then

$$\delta w_1 = -\frac{\Delta t}{2} \vec{u}^n \cdot \vec{\nabla} w_1^n + \frac{\Delta t}{2} q_1^n - \frac{\Delta t}{2} \frac{q_2^n}{(c^2)^n}, \quad (2.112)$$

$$\delta w_2 = -\frac{\Delta t}{2} \vec{u}^n \cdot \vec{\nabla} w_2^n + \frac{\Delta t}{4} c^n \vec{\tau} \cdot \vec{\nabla} (w_3 + w_4)^n + \frac{\Delta t}{2} (q_2^n \eta_y - q_3^n \eta_x), \quad (2.113)$$

$$\delta w_3 = -\frac{\Delta t}{2} (\vec{u} + c \vec{\eta})^n \cdot \vec{\nabla} w_3^n + \frac{\Delta t}{2} c^n \vec{\tau} \cdot \vec{\nabla} w_2^n + \frac{\Delta t}{2} \left(q_2^n \eta_x + q_3^n \eta_y + \frac{q_4^n}{\rho^n c^n} \right), \quad (2.114)$$

$$\delta w_4 = -\frac{\Delta t}{2} (\vec{u} - c \vec{\eta})^n \cdot \vec{\nabla} w_4^n + \frac{\Delta t}{2} c^n \vec{\tau} \cdot \vec{\nabla} w_2^n - \frac{\Delta t}{2} \left(q_2^n \eta_x + q_3^n \eta_y - \frac{q_4^n}{\rho^n c^n} \right). \quad (2.115)$$

As described in the previous section, δ in equation (2.88) refers to increments in either time or space along the characteristic directions. Let δ in equations (2.112) - (2.115) refer to the change in a characteristic variable from time t^n to $t^{n+\frac{1}{2}}$. The gradient of the characteristic variables are also defined by equation (2.88) where δ refers to spatial increments. For the above system of characteristic equations, the gradients of the characteristic variables at time t^n , found by employing equation (2.102), are defined as

$$\vec{\nabla} w_1^n = \vec{\nabla} \rho^n - \frac{\vec{\nabla} P^n}{(c^2)^n}, \quad (2.116)$$

$$\vec{\nabla} w_2^n = -\vec{\nabla} u_\tau^n, \quad (2.117)$$

$$\vec{\nabla} w_3^n = \vec{\nabla} u_\eta^n + \frac{\vec{\nabla} P^n}{\rho^n c^n}, \quad (2.118)$$

$$\vec{\nabla} w_4^n = -\vec{\nabla} u_\eta^n + \frac{\vec{\nabla} P^n}{\rho^n c^n}. \quad (2.119)$$

Therefore at time t^n , the spatial increments of the characteristic variables can be defined as the linear combination of the flow variable gradients. The finite element discretizations of the spatial increments of the characteristic variables are also then a linear combination of the finite element discretizations of the flow variables. Using the finite element approximations found in the first step of the two-step Taylor-Galerkin FEM scheme, the spatial finite element discretizations of equations (2.116) - (2.119) for an element e with a face on the boundary are

$$[\vec{\nabla} N]_e w_1^n = [\vec{\nabla} N]_e \{\rho^n\}_e - \frac{[\vec{\nabla} N]_e \{P^n\}_e}{\bar{c}^2}, \quad (2.120)$$

$$[\vec{\nabla} N]_e w_2^n = -[\vec{\nabla} N]_e \{u_\tau^n\}_e, \quad (2.121)$$

$$[\vec{\nabla} N]_e w_3^n = [\vec{\nabla} N]_e \{u_\eta^n\}_e + \frac{[\vec{\nabla} N]_e \{P^n\}_e}{\bar{\rho} \bar{c}}, \quad (2.122)$$

$$[\vec{\nabla} N]_e w_4^n = -[\vec{\nabla} N]_e \{u_\eta^n\}_e + \frac{[\vec{\nabla} N]_e \{P^n\}_e}{\bar{\rho} \bar{c}}. \quad (2.123)$$

Substituting these expressions into equations (2.112) - (2.115) yields the elemental change in the characteristic variables from t^n to $t^{n+\frac{1}{2}}$,

$$\delta_e w_1 = -\frac{\Delta t}{2} \bar{\vec{u}} \cdot [\vec{\nabla} N]_e w_1^n + \frac{\Delta t}{2} q_1^n - \frac{\Delta t}{2} \frac{q_2^n}{\bar{c}}, \quad (2.124)$$

$$\delta_e w_2 = -\frac{\Delta t}{2} \bar{\vec{u}} \cdot [\vec{\nabla} N]_e w_2^n + \frac{\Delta t}{4} \bar{c} \bar{\vec{\tau}} \cdot [\vec{\nabla} N]_e (w_3 + w_4)^n + \frac{\Delta t}{2} (q_2^n \eta_y - q_3^n \eta_x), \quad (2.125)$$

$$\delta_e w_3 = -\frac{\Delta t}{2} (\bar{\vec{u}} + \bar{c} \bar{\vec{\eta}}) \cdot [\vec{\nabla} N]_e w_3^n + \frac{\Delta t}{2} \bar{c} \bar{\vec{\tau}} \cdot [\vec{\nabla} N]_e w_2^n + \frac{\Delta t}{2} \left(q_2^n \eta_x + q_3^n \eta_y + \frac{q_4^n}{\bar{\rho} \bar{c}} \right), \quad (2.126)$$

$$\delta_e w_4 = -\frac{\Delta t}{2} (\bar{\vec{u}} - \bar{c} \bar{\vec{\eta}}) \cdot [\vec{\nabla} N]_e w_4^n + \frac{\Delta t}{2} \bar{c} \bar{\vec{\tau}} \cdot [\vec{\nabla} N]_e w_2^n - \frac{\Delta t}{2} \left(q_2^n \eta_x + q_3^n \eta_y - \frac{q_4^n}{\bar{\rho} \bar{c}} \right). \quad (2.127)$$

In equations (2.120) - (2.127), the overbar on the coefficients that are in terms of the flow variables denote state variables at time t^n . These state variables are boundary face averages. The terms containing finite element gradients are elemental constants for linear triangular finite elements. Thus, an elemental gradient holds at a boundary face of an element. Note that $\bar{\vec{\eta}}$ and $\bar{\vec{\tau}}$ are defined as constants along a boundary face of an element. Determination of $\bar{\vec{\eta}}$ for a boundary face is illustrated in Appendix A. The source terms q^n are also defined as elemental constants. The end result is that the elemental change in the characteristic variables, denoted by δ_e , also holds on the boundary face of an element.

The procedure developed here is to apply δ_e for the characteristic variables at the boundary combined with the physical boundary conditions to recover the flow variables at the boundary nodes. That is, δ_e is used for the spatially constant incremental change in the characteristic variables on the boundary face in conjunction with the physical boundary conditions to determine the spatially constant incremental change in the flow variables on the boundary face. These incremental changes in the flow variables on the boundary face are added to the flow variables at time t^n on the nodes defining the boundary face to give boundary node values of the flow variables at time $t^{n+\frac{1}{2}}$. This results in a linear distribution of the flow variables along the boundary face with which to construct the convective fluxes for the convective flux boundary integral of equation (2.12). A linear variation of the flow variables along the boundary face gives superior simulation results for the two-step Taylor-Galerkin FEM scheme compared to constant boundary face values of the convective flux. Thus, the convective fluxes in the boundary integral of equation (2.12) must be discretized with linear shape functions N

instead of constant shape functions P_e . Therefore, the convective flux boundary integral, found in equations (2.12) and (2.15), is replaced with

$$\int_{\Gamma} \{N\}[N] d\Gamma \vec{\eta} \cdot \{\vec{F}^{n+\frac{1}{2}}\}$$

so that equation (2.12) becomes

$$\begin{aligned} \sum_e \int_{\Omega} \{N\}[N] d\Omega \{\delta U\} = & \Delta t \sum_e \int_{\Omega} \{\vec{\nabla} N\} d\Omega \cdot \vec{F}_e^{n+\frac{1}{2}} - \Delta t \sum_e \int_{\Gamma} \{N\}[N] d\Gamma \vec{\eta} \cdot \{\vec{F}^{n+\frac{1}{2}}\} \\ & - \Delta t \sum_e \int_{\Omega} \{\vec{\nabla} N\} d\Omega \cdot \vec{F}_v^n + \Delta t \sum_e \int_{\Gamma} \{N\} d\Gamma \vec{\eta} \cdot \vec{F}_v^n + \Delta t \sum_e \int_{\Omega} \{N\} d\Omega Q_e^{n+\frac{1}{2}}, \end{aligned} \quad (2.128)$$

and similarly for equation (2.15),

$$\begin{aligned} R = & \Delta t \sum_e \int_{\Omega} \{\vec{\nabla} N\} d\Omega \cdot \vec{F}_e^{n+\frac{1}{2}} - \Delta t \sum_e \int_{\Gamma} \{N\}[N] d\Gamma \vec{\eta} \cdot \{\vec{F}^{n+\frac{1}{2}}\} - \Delta t \sum_e \int_{\Omega} \{\vec{\nabla} N\} d\Omega \cdot \vec{F}_v^n \\ & + \Delta t \sum_e \int_{\Gamma} \{N\} d\Gamma \vec{\eta} \cdot \vec{F}_v^n + \Delta t \sum_e \int_{\Omega} \{N\} d\Omega Q_e^{n+\frac{1}{2}}. \end{aligned} \quad (2.129)$$

Evaluation of the boundary and volume integrals found in equations (2.128) and (2.129) is described in Appendix A.

There are two important points to be made about the numerical treatment of equations (2.124) - (2.127). Consider Figure 2-4 which shows a portion of an unstructured triangular finite element mesh in the vicinity of a boundary. Element E is defined by nodes 1, 2, and 3. Γ defines the boundary of the domain and $\vec{\eta}$ is the unit outward normal corresponding to the boundary face of element E defined by nodes 1 and 2. The common procedure for determining the numerical boundary conditions on structured grids is to perform space and time extrapolation of either the flow variables or characteristic variables from interior points (Hirsch, 1990). The first several computational points in from the boundary are aligned in the direction normal to the boundary surface. Then, for a second-order accurate spatial scheme, first-order extrapolation of the variables is performed. This is an acceptable procedure based upon the theorem of Gustafsson (1975) where, for linear equations, the boundary scheme can be one order lower than the interior scheme without reducing the global order of accuracy of the overall scheme. This theorem is based on linearized theory, but has been confirmed by numerical tests on nonlinear equations such as the Euler equations. Another possible procedure is to employ one-sided differencing of the characteristic equations in the normal directions to the boundary, such as described by Chakravarthy (1983). However, this procedure also requires interior point alignment with the boundary point in the normal direction. Considering the finite element mesh in Figure 2-4, there is no readily apparent way to implement an extrapolation procedure in the normal direction $\vec{\eta}_{\Gamma}$ on this mesh. Nodes would also have to be aligned normal to the boundary nodes. This might be possible but

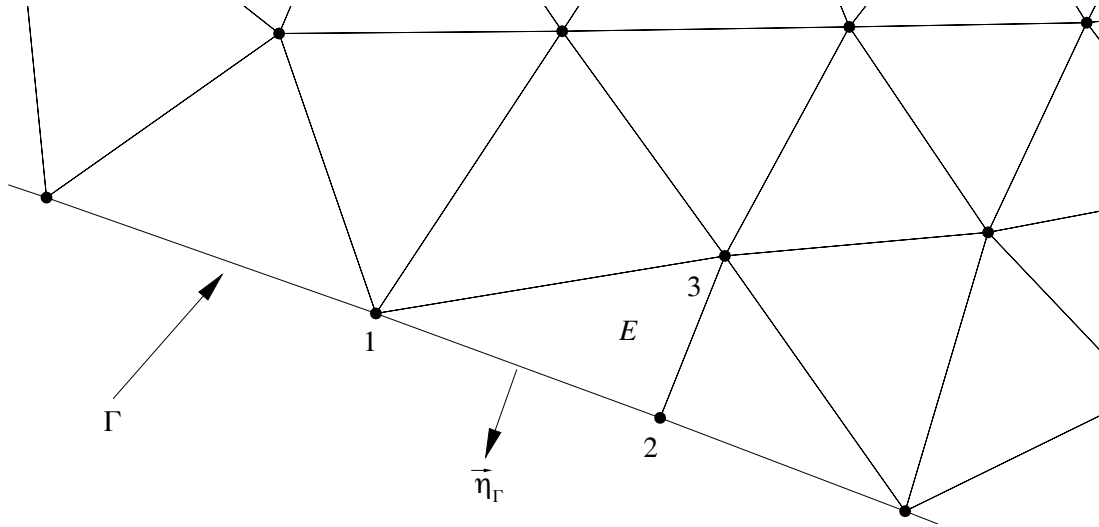


Figure 2-4 A portion of an unstructured triangular finite element mesh near a boundary.

would also change the finite element formulation as $\vec{\eta}$ and $\vec{\tau}$ would be defined at the nodes instead of being uniquely defined on a boundary face. Fortunately, the FEM offers much simpler procedures to maintain spatial accuracy on the boundary. The spatial derivatives for linear finite elements are spatially second-order accurate and, therefore, there is no need to incorporate information from interior elements. The procedure here incorporates spatial derivatives from the elements adjacent to the boundary only. The derivatives found in equations (2.124) - (2.127) maintain the same spatial accuracy as the two-step Taylor-Galerkin FEM scheme.

The second point of interest is that equations (2.124) - (2.127) are formulated in an incremental fashion. At no point are the characteristic variables defined at times t^n to $t^{n+\frac{1}{2}}$, only their spatial and temporal increments are defined. In their final discretized form, equations (2.124) - (2.127) could conceivably be integrated in time to find the values of the characteristic variables at time $t^{n+\frac{1}{2}}$. However, this would require knowledge of the values of the characteristic variables at time t^n . The characteristic paths intersecting the boundary at time $t^{n+\frac{1}{2}}$ would have to be traced back to the spatial point of departure at time t^n . Given that the characteristic paths are defined by equation (2.83) in the normal direction $\vec{\eta}$, there may be multiple departure points corresponding to the different positive eigenvalues. For illustration purposes, consider the portion of a one-dimensional grid near a supersonic exit, shown in Figure 2-5, where all of the characteristic paths exit the boundary. Element E is defined by boundary node i and interior node $i-1$. In order to find the values of the characteristic variables at time $t^{n+\frac{1}{2}}$ coinciding at the spatial position of boundary node i , each characteristic path corresponding to a positive eigenvalue (in this case $\lambda_{1,4}$) would have to be traced back to the spatial point of departure, where the characteristic path intersects the x -axis at time t^n . Unfortunately, none of the values of the characteristic variables at time t^n or $t^{n+\frac{1}{2}}$ are known. However, the temporal increment of the flow variables can be recovered from the temporal

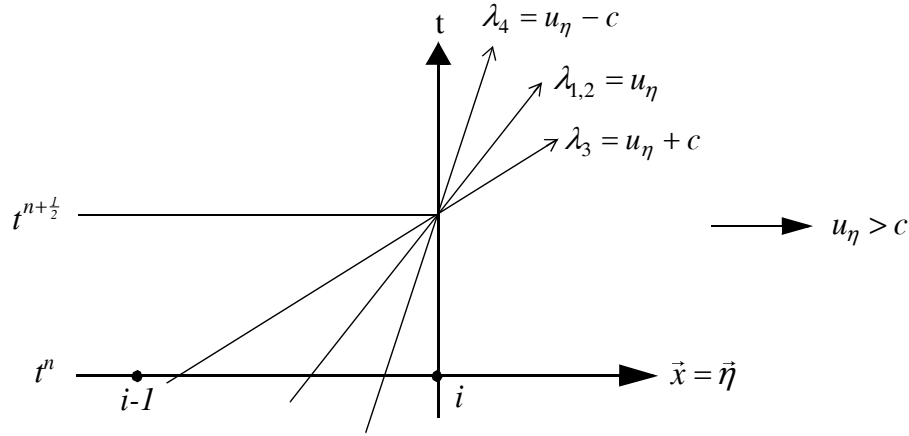


Figure 2-5 One-dimensional characteristic paths for a supersonic flow exit.

increment of the characteristic variables at the boundary with the information provided by the physical boundary conditions. In the following sections, this procedure will be illustrated for several boundary condition types.

2.5.3 Free-Slip Solid Wall and Symmetry Boundary Conditions

For free-slip solid wall and symmetry boundaries, the physical boundary condition to be specified is that the normal component of velocity is suppressed, $u_\eta = 0$. The conservative flux vector in the outward normal direction, equation (2.111), reduces to

$$\vec{\eta} \cdot \vec{F}_e^{n+\frac{1}{2}} = \begin{Bmatrix} 0 \\ P\eta_x \\ P\eta_y \\ 0 \end{Bmatrix}^{n+\frac{1}{2}}.$$

Therefore, only the determination of the boundary pressure is required for free-slip solid wall and symmetry boundaries. The characteristic directions for this type of boundary is illustrated in Figure 2-1. The eigenvalue λ_4 is negative and represents a characteristic wave entering the domain that must be replaced by the physical boundary condition, $u_\eta = 0$. One characteristic is exiting the domain, defined by λ_3 . Therefore, equation (2.126) must be solved to determine the information propagating from the interior of the domain. Applying $u_\eta = 0$ to the definition of the change in w_3 , equation (2.102), yields

$$\delta_e w_3 = \frac{\delta P}{\rho c}.$$

With the state variable $\bar{\rho c}$ known from the boundary face averaged flow variables at time t^n and δw_λ determined from equation (2.126), δP is then solved for directly. The pressure at time $t^{n+\frac{1}{2}}$ for the boundary nodes 1 and 2 of element E (see Figure 2-4) is then

$$\begin{aligned} P_1^{n+\frac{1}{2}} &= P_1^n + \delta P_1 \\ P_2^{n+\frac{1}{2}} &= P_2^n + \delta P_2 . \end{aligned} \quad (2.130)$$

For this type of boundary condition, flow tangency at the boundary must also be strongly enforced. With the normal velocity specified to be zero, the operation for rotating the velocity vector tangent with the boundary,

$$\vec{u}_\tau = [I - \eta\eta^T] \vec{u}, \quad (2.131)$$

is applied to the momentum components of the residual R in equation (2.129) and any diffusion added to the momentum components on the boundary (Argyris, 1989). For this operation, \vec{u} represents any vector where the normal component is to be suppressed. \vec{u}_τ is the result after the operation.

2.5.4 Characteristic Boundary Conditions for Flow Exits

Two types of flow exits will be addressed here. The first is a subsonic flow exit with the pressure at the boundary specified, termed back pressure. The second type of flow exit is the nonreflective flow exit. In the subsonic non-reflective case, the physical boundary condition to be specified is that no waves be propagated along the incoming characteristic. A supersonic exit is inherently non-reflective because all of the characteristics exit the flow boundary and there is no incoming information. In other words, acoustic signals cannot propagate upstream in a supersonic flow. Figure 2-2 illustrates the characteristic directions for both subsonic and supersonic flow exits.

2.5.4.1 Subsonic Flow Exit with Specified Back Pressure

Figure 2-2a shows that for a subsonic exit there is one characteristic entering the domain, λ_4 . For the case of specified back pressure, the characteristic defined by λ_4 is replaced by the specified back pressure. The characteristics defined by λ_1 , λ_2 , and λ_3 are exiting the boundary. The information for these characteristics are obtained by solving equations (2.124) - (2.126).

With the numerical boundary conditions $\delta_e w_1$, $\delta_e w_2$, and $\delta_e w_3$ known along the boundary face by equations (2.124) - (2.126) and δP being specified at the boundary nodes, the other flow

variables are easily obtained for boundary nodes 1 and 2 (see Figure 2-4) using equation (2.102). Density at the nodes is found by

$$\begin{aligned}\rho_1^{n+\frac{1}{2}} &= \rho_1^n + \delta_e w_1 + \frac{\delta P_1}{\bar{c}^2} \\ \rho_2^{n+\frac{1}{2}} &= \rho_2^n + \delta_e w_1 + \frac{\delta P_2}{\bar{c}^2},\end{aligned}\tag{2.132}$$

the tangential velocity component at the boundary nodes is determined by

$$\begin{aligned}u_{\tau 1}^{n+\frac{1}{2}} &= u_{\tau 1}^n + \delta_e w_2 \\ u_{\tau 2}^{n+\frac{1}{2}} &= u_{\tau 2}^n + \delta_e w_2,\end{aligned}\tag{2.133}$$

and the normal velocity component at the boundary nodes is determined by

$$\begin{aligned}u_{\eta 1}^{n+\frac{1}{2}} &= u_{\eta 1}^n + \delta_e w_3 - \frac{\delta P_1}{\rho \bar{c}} \\ u_{\eta 2}^{n+\frac{1}{2}} &= u_{\eta 2}^n + \delta_e w_3 - \frac{\delta P_2}{\rho \bar{c}}.\end{aligned}\tag{2.134}$$

The temperature at the boundary nodes can be determined with the use of equation (2.130) and by the equation of state, equation (1.3),

$$\begin{aligned}T_1^{n+\frac{1}{2}} &= \frac{P_1^{n+\frac{1}{2}}}{R_c \rho_1^{n+\frac{1}{2}}} \\ T_2^{n+\frac{1}{2}} &= \frac{P_2^{n+\frac{1}{2}}}{R_c \rho_2^{n+\frac{1}{2}}}.\end{aligned}\tag{2.135}$$

2.5.4.2 Non-Reflecting Flow Exit

Non-reflecting (or transmissive) boundary conditions are useful in that they allow the CFD user to find a convenient point at which to place a flow exit boundary instead of having to extend the grid far downstream in order to avoid undesired acoustic waves from propagating upstream into the region of the domain. The subsonic non-reflecting boundary condition expresses the physical boundary condition as the requirement that the local perturbations propagated along incoming characteristics be made to vanish (Hirsch, 1990). This subsonic non-reflective requirement can be expressed in equation (2.97) as

$$\frac{\partial W}{\partial t} = L^{-1} \tilde{Q}.\tag{2.136}$$

Equation (2.136) is the physical boundary condition that replaces the incoming characteristic defined by λ_4 . From equation (2.127), the resulting characteristic equation with the subsonic non-reflective requirement for λ_4 is then

$$\delta_e w_4 = \frac{\Delta t}{2} \bar{c} \vec{\tau} \cdot [\vec{\nabla} N]_e w_2^n - \frac{\Delta t}{2} \left(q_2^n \eta_x + q_3^n \eta_y - \frac{q_4^n}{\rho c} \right). \quad (2.137)$$

Therefore, the numerical boundary conditions are obtained with equations (2.124) - (2.126) and the physical boundary condition is obtained with equation (2.137). Solution of these equation will give $\delta_e w_1$, $\delta_e w_2$, $\delta_e w_3$, and $\delta_e w_4$ at the boundary. The result is four equations and four unknowns.

Simultaneous solution of all of the incremental changes in characteristic variables found in equation (2.102) gives the required flow variables at the flow exit boundary. The incremental change in pressure at the boundary nodes is found by

$$\delta P_1 = \delta P_2 = \frac{\bar{\rho} c}{2} (\delta_e w_3 + \delta_e w_4). \quad (2.138)$$

Using equation (2.130) with equation (2.138) gives the boundary node pressures. As in the case of a subsonic exit with specified back pressure, density is found at the nodes by employing equation (2.132). With pressure and density now known at the boundary, the temperatures at the boundary nodes are determined with equation (2.135). The normal velocity components at the boundary nodes are determined by

$$u_{\eta 1}^{n+\frac{1}{2}} = u_{\eta 1}^n + \frac{\delta_e w_3 - \delta_e w_4}{2} \quad (2.139)$$

$$u_{\eta 2}^{n+\frac{1}{2}} = u_{\eta 2}^n + \frac{\delta_e w_3 - \delta_e w_4}{2}.$$

The tangential velocity components and the temperatures at the boundary nodes are found as before with equations (2.133) and (2.135), respectively.

For supersonic exits, the boundary conditions are inherently non-reflective because all of the characteristics exit the flow boundary. The same procedure for determining the flow variables at the boundary is applied for the supersonic case as the subsonic case. However, the characteristic defined by λ_4 is now exiting the boundary. Therefore, equation (2.127) is used in place of equation (2.137).

2.5.5 Characteristic Boundary Conditions for Flow Inlets

Referring to Figure 2-3, characteristic boundary conditions are only necessary for subsonic inlet flow conditions. Figure 2-3a shows that three physical boundary conditions are required and only one numerical boundary condition corresponding to λ_3 is required. Supersonic inlet boundary conditions do not require the application of characteristic equations as all of the characteristics are defined by in-coming waves as shown in Figure 2-3b. Therefore, all variables at the inlet boundary must be specified resulting in four physical boundary conditions and no numerical boundary conditions. The three flow inlet boundary condition types presented below are specified boundary pressure and temperature, specified stagnation pressure and temperature, and mass injection. In all three cases, the direction of inflow must be specified. For mass injection, this is accomplished by the specification of the velocity components.

2.5.5.1 Subsonic Inlet with Specified Boundary Pressure and Temperature

The physical boundary conditions for a subsonic inlet with specified boundary pressure and temperature is the boundary pressure and temperature and the direction of inflow defined by a user supplied unit vector \vec{t} with components in the x and y directions, t_x and t_y , respectively. With λ_3 being positive, the numerical boundary condition comes from the solution of equation (2.126) to provide $\delta_e w_3$. δP is known on the boundary because of the specified boundary pressure. Also, ρ and c are known on the boundary due to the specified boundary temperature. Thus, the normal velocity component $u_\eta^{n+\frac{1}{2}}$ may be directly calculated for the boundary nodes from equation (2.134). The magnitude of velocity for the boundary nodes can be determined by first finding the projection of the specified direction of flow upon the outward normal direction of flow,

$$\theta = |\vec{t} \cdot \vec{\eta}|, \quad (2.140)$$

and then solving for the magnitude of velocity at the boundary nodes from the projection in the normal direction,

$$\begin{aligned} |\vec{u}_1^{n+\frac{1}{2}}| &= \frac{|u_\eta^{n+\frac{1}{2}}|}{\theta} \\ |\vec{u}_2^{n+\frac{1}{2}}| &= \frac{|u_{\eta 2}^{n+\frac{1}{2}}|}{\theta}. \end{aligned} \quad (2.141)$$

The components of velocity at the boundary node 1 are then found directly from

$$\begin{aligned} u_1^{n+\frac{1}{2}} &= \iota_x \left| \vec{u}_1^{n+\frac{1}{2}} \right| \\ v_1^{n+\frac{1}{2}} &= \iota_y \left| \vec{u}_1^{n+\frac{1}{2}} \right|, \end{aligned} \tag{2.142}$$

and for boundary node 2,

$$\begin{aligned} u_2^{n+\frac{1}{2}} &= \iota_x \left| \vec{u}_2^{n+\frac{1}{2}} \right| \\ v_2^{n+\frac{1}{2}} &= \iota_y \left| \vec{u}_2^{n+\frac{1}{2}} \right|. \end{aligned} \tag{2.143}$$

2.5.5.2 Subsonic Inlet with Specified Stagnation Pressure and Temperature

The application of characteristic boundary conditions for a subsonic flow inlet with specified stagnation pressure P_o and temperature T_o is similar to the specified boundary pressure and temperature case. However, because the boundary pressure and temperature are now functions of the stagnation pressure and temperature and fluid velocity at the boundary, the application of characteristic boundary conditions is considerably more difficult. Fortunately, a formulation that greatly simplifies the application and has shown to improve the simulation performance can be found in the non-reflective requirement for characteristic equations. As described in Section 2.5.4.2, the non-reflective requirement may be chosen to replace a physical boundary condition. Because boundary velocity is needed to compute the thermodynamic flow variables at the boundary, it is convenient to replace the physical boundary condition associated with a negative λ_4 with equation (2.137) to obtain $\delta_e w_4$. Combined with $\delta_e w_3$, the normal velocity component $u_\eta^{n+\frac{1}{2}}$ for the boundary nodes is determined from equation (2.139). The procedure to find the magnitude of velocity and velocity components on the boundary is then identical to the specified pressure and temperature application of the preceding section, equations (2.140) - (2.143). This non-reflective procedure for obtaining velocity at the boundary nodes provides superior performance over the more complex treatment (not presented here) in steady-state simulations. Waves propagating upstream to the boundary are damped, which results in faster convergence for steady-state simulations.

After following the above non-reflective procedure to obtain the components of velocity at the boundary nodes, the boundary temperature, pressure, and density are then determined from

the equation of state. Rearranging equation (1.7) to solve for temperature at the boundary nodes yields

$$T_1^{n+\frac{1}{2}} = T_o - \frac{|\vec{u}_1^{n+\frac{1}{2}}|^2}{2c_p} \quad (2.144)$$

$$T_2^{n+\frac{1}{2}} = T_o - \frac{|\vec{u}_2^{n+\frac{1}{2}}|^2}{2c_p}.$$

Using these boundary node values of temperature and solving equation (1.9) for boundary node pressure,

$$P_1^{n+\frac{1}{2}} = P_o \left(\frac{T_o}{T_1^{n+\frac{1}{2}}} \right)^{\frac{-\gamma}{\gamma-1}} \quad (2.145)$$

$$P_2^{n+\frac{1}{2}} = P_o \left(\frac{T_o}{T_2^{n+\frac{1}{2}}} \right)^{\frac{-\gamma}{\gamma-1}}.$$

And finally, solving for the boundary node values of density from equation (1.3) gives

$$\rho_1^{n+\frac{1}{2}} = \frac{P_1^{n+\frac{1}{2}}}{R_c T_1^{n+\frac{1}{2}}} \quad (2.146)$$

$$\rho_2^{n+\frac{1}{2}} = \frac{P_2^{n+\frac{1}{2}}}{R_c T_2^{n+\frac{1}{2}}}.$$

2.5.5.3 Mass Injection

The physical boundary conditions for mass injection are the components of velocity and density which is the same as specified mass flow rate and direction. As before this is a subsonic flow inlet. Thus, λ_3 is positive and the numerical boundary condition must come from the solution of equation (2.126) for $\delta_e w_3$ to provide the information propagating from the interior of the domain. Because the normal velocity is specified and the state variable is

known, the expression for $\delta_e w_3$, equation (2.102), can be rearranged to give pressure at the boundary nodes,

$$P_1^{n+\frac{1}{2}} = P_1^n + \overline{\rho c}(\delta_e w_3 - u_{\eta 1}^{n+\frac{1}{2}}) \quad (2.147)$$

$$P_2^{n+\frac{1}{2}} = P_2^n + \overline{\rho c}(\delta_e w_3 - u_{\eta 2}^{n+\frac{1}{2}}).$$

Temperature at the boundary nodes is then determined by the equation of state, equation (2.135).

2.5.6 Viscous Boundary Conditions

As discussed in Chapter 1, the coupled system of the time-dependent Navier-Stokes equations is a hybrid system, being parabolic-hyperbolic for transients but becoming mixed elliptic-hyperbolic for the steady-state formulation. It is noted that, while irrotational flow is easily maintained at a flow inlet because either the velocity components or the flow direction is specified, there is generally viscous behavior at the exit of the flow. Wakes due to flow around a solid body, slip lines, and shocks can result in viscous shear at a flow exit. In nearly all viscous flows, the treatment of a viscous boundary layer must be considered. The viscous boundary conditions must also be compatible with inviscid conditions in the limit of vanishing viscosity.

Because the viscous effects cannot be ignored or minimized near a no-slip solid wall, the characteristic equations are not appropriate. This point is of limited importance because the viscous boundary integral in equation (2.129) for the momentum components is not needed because all momentum components are zero at a no-slip solid wall. For the total energy equation at a no-slip solid wall, there is the requirement of heat flux in the boundary integral, but this is user defined. Also, if the temperature at the no-slip stationary solid wall is known, the stagnation enthalpy is known and, thus, the total energy is known. Therefore, the evaluation of the boundary integral for the total energy equation is unnecessary. The viscous boundary integral in equation (2.129) is only evaluated on flow inlets and exits. The approach here is to apply the same elemental viscous flux \vec{F}_v^n on the boundary face of the element.

2.5.7 Simulation Examples with Strong Flow Interaction with the Boundary

In this section, two inviscid flow examples are illustrated that test the above characteristic boundary condition method for robustness and accuracy. The first is a steady-state Mach 2.5 channel with a flow obstruction created by a 5 degree double wedge on the lower wall. It is designed to create an oblique shock wave interaction with the upper free-slip solid wall, very near a supersonic exit (non-reflective boundary). The second example is a transient Mach reflection of a shock wave from a 25 degree wedge. The flow is entirely subsonic but contains

a plane shock wave propagating left to right at Mach 1.3. Both examples are solved with the two-step Taylor-Galerkin FEM with FCT as the high-resolution scheme.

2.5.7.1 Steady Mach 2.5 Channel with a 5 Degree Double Wedge Obstruction

The first simulation example is of an inviscid Mach 2.5 channel at steady-state with a flow obstruction created by a 5 degree double wedge on the lower wall. The entire flow field for this simulation is supersonic and thus requires that all flow variables be specified at the inlet and that nothing may be specified at the flow exit. Figure 2-6 shows the geometry of the

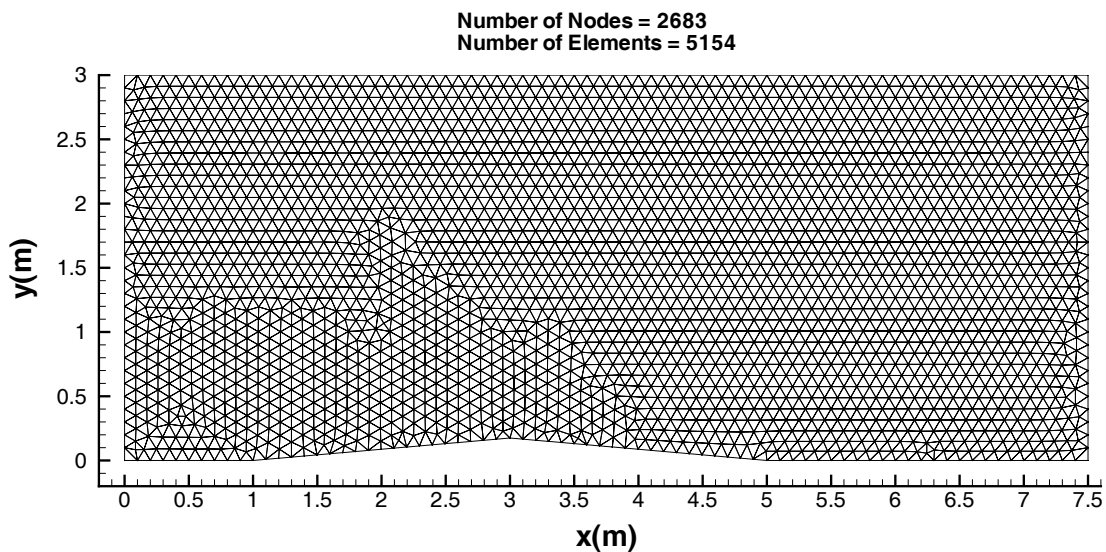


Figure 2-6 Domain geometry and triangular finite element mesh for Mach 2.5 channel with 5 degree double wedge channel obstruction.

domain discretized with 2683 nodes and 5154 linear triangular elements. The flow inlet conditions are applied at the left boundary, $x = 0$ m, with the inlet pressure and temperature specified to be $P = 101,325$ Pa and $T = 300$ K, respectively. The inlet velocity components corresponding to a Mach 2.5 flow are $u = 868.032$ m/s and $v = 0.0$ m/s. For the domain exit, the right boundary at $x = 7.5$ m, the values of the flow variables are determined using the method outlined in Section 2.5.4.2 for the non-reflecting flow exit. The double wedge extends along the bottom boundary from $x = 1$ m to $x = 5$ m with the apex of the channel obstruction at $x = 3$ m. The top boundary is constant at $y = 3$ m. Both the bottom and top boundaries are prescribed as free-slip solid walls. At these boundaries, the procedure described in Section 2.5.3 is applied.

Figure 2-7 graphically illustrates the numerical steady-state pressure solution for the Mach 2.5 channel problem. The solution results in a double oblique shock wave and expansion wave system. The leading edge of the double wedge generates the first shock wave and the trailing

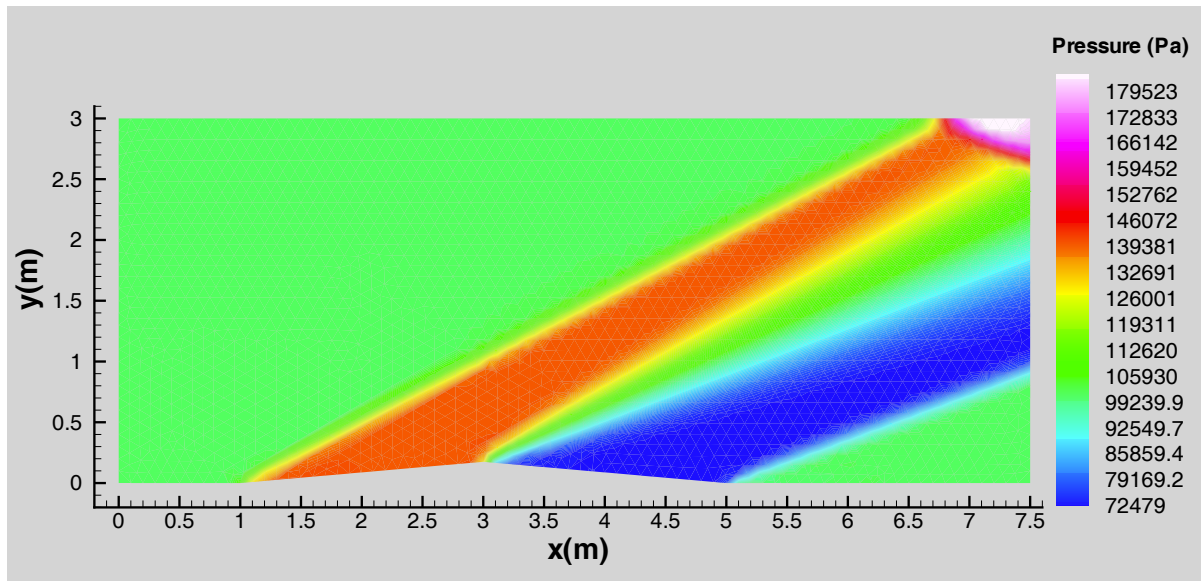


Figure 2-7 Steady-State numerical pressure solution for Mach 2.5 channel.

edge generates the second shock wave. In between are two constant state regions and an expansion fan. Using the shock wave tables of NACA Report 1135 (1953), the leading edge of the 5 degree double wedge will generate an oblique shock at an angle of approximately 27.5 degrees. For the domain employed here, this first shock impacts the top boundary at approximately $x = 6.75$ m. This impact creates a reflected shock that immediately exits the boundary. This reflected shock is also interacting with the expansion fan in the region near $x = 7.1$ m to $x = 7.5$ m and $y = 2.4$ m to $y = 2.7$ m. This shock interaction results in difficult flow conditions near the exit boundary that, in order to be accurately modeled, requires rigorous treatment of boundary conditions. Figure 2-8 is a magnification of the pressure solution in this region. It shows pressure contours (isobars) as they exit the domain. As desired, the pressure contours smoothly exit the boundary, showing a consistency of the boundary solution with the interior solution. The pressure contour wiggles on the shocks are primarily due to the graphics package (TECPLOT®, 1999) interpolation of the pressure across the linear triangles and are not the result of numerical oscillations.

2.5.7.2 Transient Mach Reflection of a Shock Wave from a 25 Degree Wedge

The second simulation example is the Mach reflection of a plane shock wave from a 25 degree wedge. Where in the previous simulation example, the flow was entirely supersonic, the flow field for this problem is entirely subsonic. It is a transient problem in which a plane shock wave is propagating left to right at a Mach number of 1.3 and obliquely impacts a 25 degree wedge. This impact creates a normal shock, or Mach stem, perpendicular to the wedge. Where the Mach stem intersects the incoming plane shock wave, a resulting slip line is formed. The intersection of these three flow features is termed a triple point. The primary purpose of this simulation is to test how accurately the characteristic boundary condition procedure described

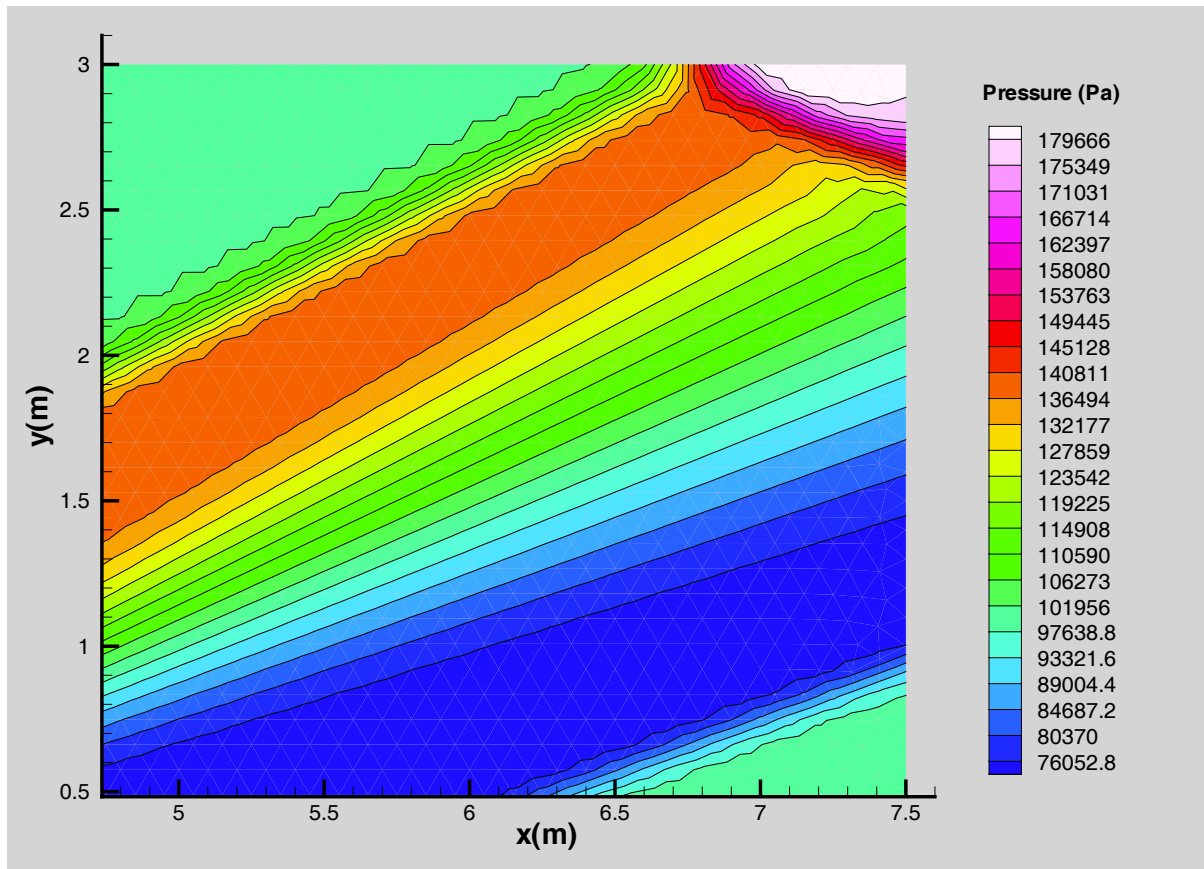


Figure 2-8 Magnified region of oblique shock wave interaction with the exit boundary for the Mach 2.5 channel.

in the preceding sections can compute the flow inlet velocity boundary condition and how well the free-slip solid wall boundary condition of the wedge generates the Mach stem.

Figure 2-9 shows the geometry of the domain discretized with 6706 nodes and 13,104 linear triangular elements. The wedge extends along the bottom boundary from $x = 1.0$ m to $x = 2.0$ m. The flow inlet boundary is located at the left boundary, $x = 0.0$ m. The initial conditions chosen for this simulation are $P = 101,325.0$ Pa, $T = 300.0$ K, and $u = v = 0.0$ m/s. The normal shock relations found in NACA Report 1135 (1953) are used to compute the specified pressure and temperature at the inflow boundary. For the given set of initial conditions, the normal shock relations prescribe a pressure of $P = 182,891.625$ Pa and a temperature of $T = 357.262$ K. The specified direction of flow is parallel to the x -axis, $t_x = 1.0$ and $t_y = 0.0$. The normal shock relations for these conditions give an analytical solution of the velocity at the inlet boundary to be $u = 153.575$ m/s.

Figure 2-10 graphically illustrates the numerical pressure solution at a time of $t = 2.0$ ms, just prior to the shock wave encountering the wedge. There is good agreement with the analytical plane shock wave location of $x = 0.903$ m. The numerical inlet velocity is $u = 153.65$ m/s

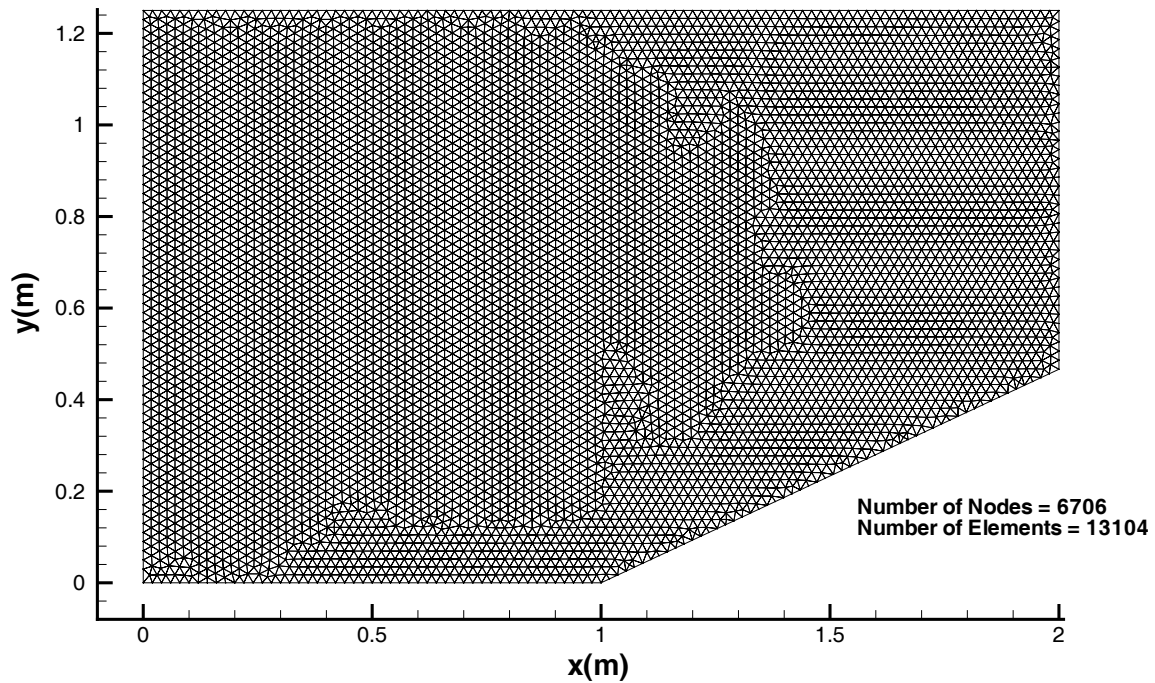


Figure 2-9 Domain geometry and triangular finite element mesh for transient Mach reflection of a shock wave from a 25 degree wedge.

(average along inlet boundary which does not vary by more than 0.1 m/s) which is very close to the analytical solution of $u = 153.575$ m/s.

Figure 2-11 shows the numerical pressure solution at a time of $t = 4.0$ ms. The Mach stem is clearly defined and smoothly intersects the wedge in the normal direction.

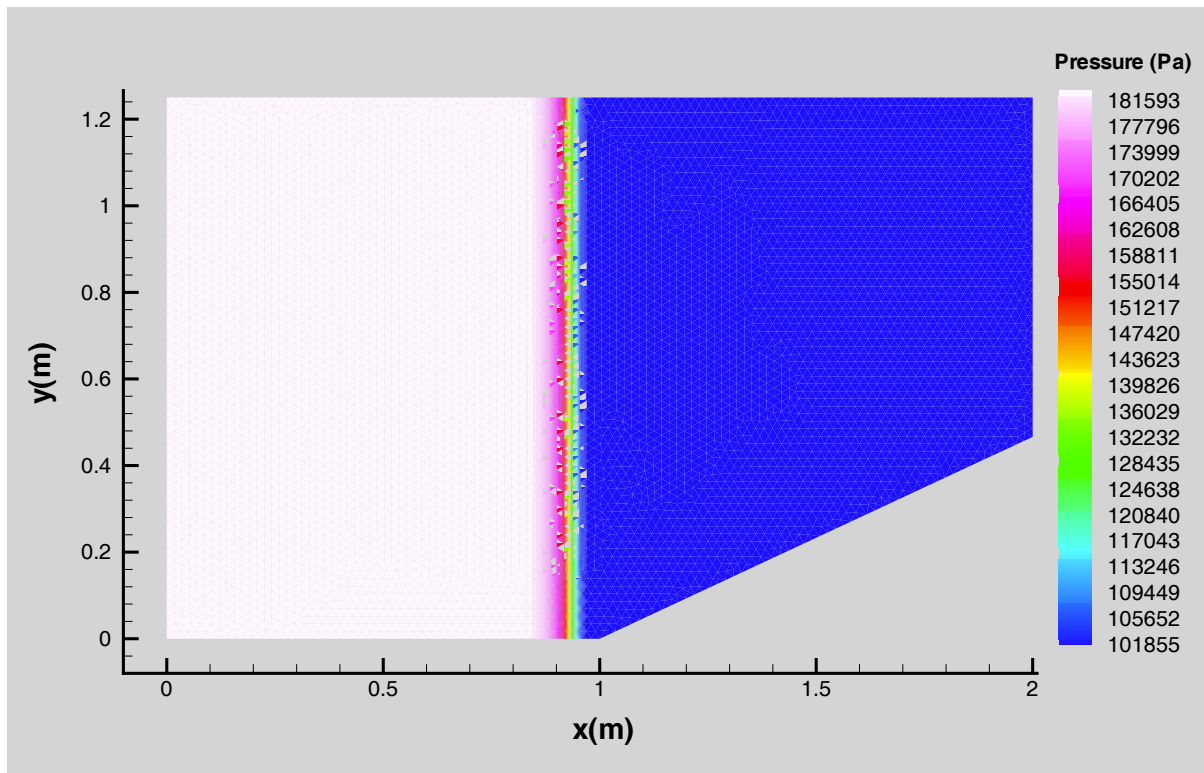


Figure 2-10 Numerical pressure solution at time $t = 2.0$ ms for 25 degree wedge simulation.

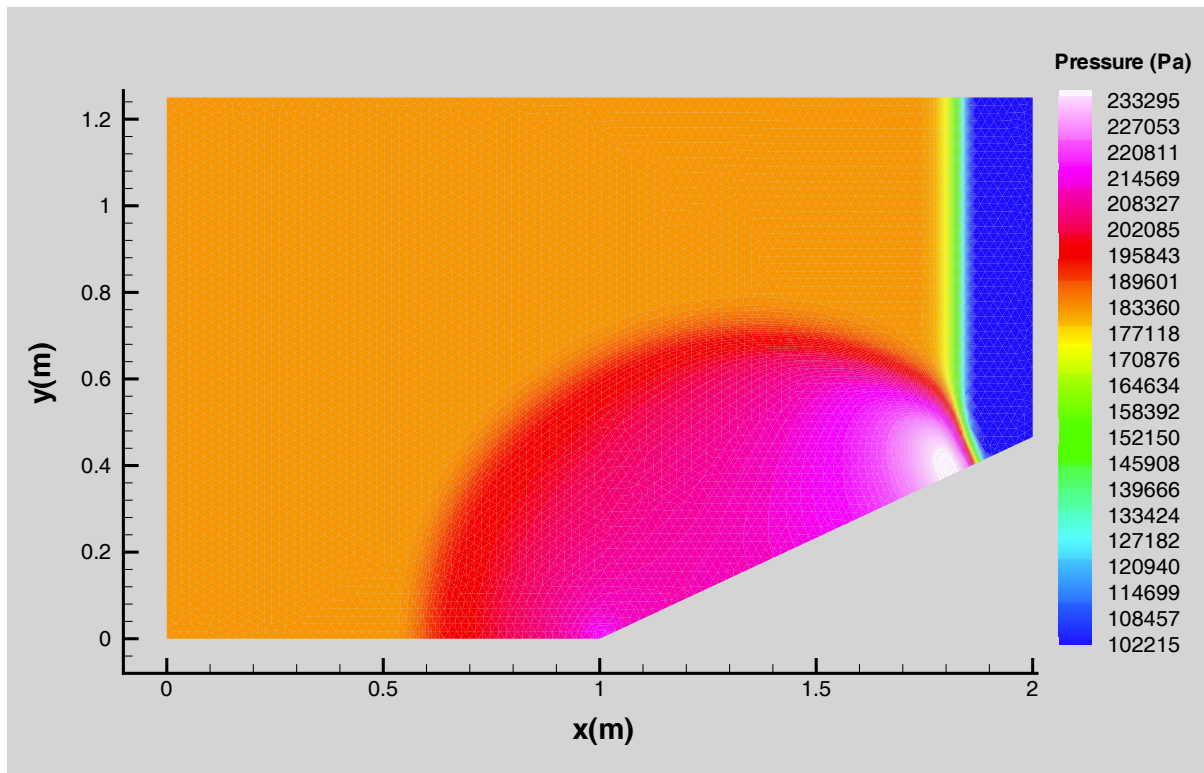


Figure 2-11 Numerical pressure solution at time $t = 4.0$ ms for 25 degree wedge simulation.

3. The Pressure-Corrected ICE Finite Element Method for Unstructured Grids (PCICE-FEM)

The primary purpose of this research is to develop a more accurate, efficient, and robust semi-implicit pressure-based computational fluid dynamics (CFD) scheme for the widest range of compressible flows. While proving the accomplishment of such an objective may be difficult, developing a superior pressure-based CFD scheme in terms of accuracy and efficiency is a reasonable proposition. This chapter is devoted to the theoretical development of a new pressure-based CFD scheme that satisfies this proposition. This new scheme is an efficient, high-resolution, high-order accurate, semi-implicit pressure-based scheme for simulating transient and steady, inviscid and viscous compressible flows on unstructured finite element meshes. Developed here, a new temporal discretization and mass-momentum coupling algorithm, termed the Pressure-Corrected Implicit Continuous-fluid Eulerian (PCICE) algorithm, is combined with the Finite Element Method (FEM) spatial discretization scheme to yield a new semi-implicit pressure-based scheme named the PCICE-FEM.

The PCICE algorithm is composed of three phases, an explicit predictor, an elliptic pressure Poisson solution, and a semi-implicit pressure-correction of the flow variables. The pressure, momentum, and density variables in the governing hydrodynamic equations are treated in an implicit fashion. Hence, the formulation is referred to as semi-implicit. The three implicit variables are directly coupled by substituting the balance of momentum equations into the mass conservation equation to eliminate momentum as an unknown. Density is then expressed in terms of pressure by employing the equation of state. This is known as mass-momentum coupling or pressure-velocity-density coupling. These substitutions result in a single second-order differential Poisson equation in terms of pressure (pressure Poisson). Thus, the scheme is termed pressure-based. The solution of the pressure Poisson equation effectively solves the mass conservation and balance of momentum equations simultaneously. This semi-implicit treatment has two advantages over explicit schemes. The first advantage is that the acoustic component from the explicit time step size stability criteria is removed. The time integration stiffness of slow flows on small computational cells, typically found in viscous boundary layer discretizations, is then eliminated. The second advantage is that the pressure obtained with this semi-implicit treatment corrects the momentum to satisfy mass conservation requirements. Theoretically, this allows incompressible flows to be simulated with compressible flow equations. This type of scheme can then be used to simulate any flow from nearly incompressible to supersonic.

The PCICE-FEM scheme incorporates a combination of the two-step Taylor-Galerkin FEM scheme and a time-weighted trapezoid rule FEM scheme for the semi-implicit equations to achieve second-order temporal differencing. Second-order spatial differencing is accomplished by linear unstructured finite element discretization. For documentation purposes, this study illustrates the PCICE-FEM development on two-dimensional flow fields. Therefore, linear triangular finite elements will be used exclusively for the finite element formulations. Linear unstructured finite elements are chosen to spatially discretize the domain for several reasons. The first is that they are easily generated on domains with complex geometries. Second, they provide second-order spatial accuracy. Third, they can be easily adapted to minimize error in the solution. And finally, they can be integrated exactly, which

eliminates the need of time consuming numerical integration. This last point is important for the PCICE-FEM scheme's efficiency. The pressure Poisson equation resulting from the mass-momentum coupling requires an iterative solver. The coefficient matrix for the pressure Poisson equation requires integration of the finite element contributions. Exact integration of these contributions is far more efficient than a numerical integration technique, such as Gauss quadrature. The PCICE algorithm could be used with other spatial differencing and high-resolution schemes. However, the explicit FEM-FCT scheme (Chapter 2) was chosen as the basis for the explicit predictor phase of PCICE-FEM scheme because of its known ability to accurately simulate strong transients on complex geometries. The PCICE-FEM scheme extends this capability. It excels on steady-state simulations, including steady-state viscous flows, and is applicable over a much wider Mach number range than the FEM-FCT scheme while maintaining the ability to accurately simulate strong transients.

The PCICE algorithm is a descendent of the Implicit Continuous-fluid Eulerian (ICE) scheme originally developed at the Los Alamos National Laboratory (LANL) by Harlow and Amsden (1971). In the original version of ICE, the mass conservation equation and the balance of momentum equations are coupled together into a Poisson equation in terms of the time-advanced pressure. The time rate of change in density term from the mass conservation equation is related to the time-advanced pressure using a barotropic like equation of state,

$$\frac{\partial P}{\partial t} = \left(\frac{\partial P}{\partial \rho} \right)_T \frac{\partial \rho}{\partial t}. \quad (3.1)$$

Equation (3.1) is equation (1.24) with the energy effects neglected. In other words, the second term on the right-hand side of equation (1.24) is ignored in the mass-momentum coupling. This mass-momentum coupling into the pressure Poisson equation effectively solves the mass conservation equation and the balance of momentum equations simultaneously. The conservation of total energy equation is then solved explicitly in time, effectively de-coupled from the conservation of mass equation and the balance of momentum equations. This early ICE scheme assumes that the change in fluid pressure is due primarily to the change in density. The change in pressure due to the changes in internal energy, temperature, or some other physical process is then assumed to be relatively small. Later ICE schemes, such as the generalized ICE method proposed by Westbrook (1978), incorporates all of the terms in the equation of state contributing to the change in pressure. The mass-momentum coupling then results in a pressure Poisson equation that contains terms requiring the relationship between pressure and energy to be known. An iterative process is utilized between the pressure Poisson equation and the implicit correction of the density, balance of momentum components, and the total energy. This process is outlined in the basic mass-momentum coupled scheme described in Section 1.2.1 where the total energy equation is in terms of time-advanced pressure, density, and momentum.

The PCICE algorithm is derived so that the total energy equation is sufficiently coupled to the pressure Poisson equation in order to avoid the iteration process between the pressure Poisson equation and the semi-implicit pressure-correction equations. A significant increase in efficiency for a semi-implicit pressure-based scheme can be obtained if the elliptic pressure

Poisson equation only has to be solved once per time step. As discussed in Section 1.2.2, direct coupling of the conservation form of the total energy with the balance of momentum equations is not possible because there is no readily apparent way to eliminate total energy as an unknown in the convective flux term of the total energy equation. Therefore, a weaker coupling of energy must be found that will still provide sufficient information to the pressure Poisson equation to yield an accurate implicit pressure field. This weaker coupling must come through the equation of state used to eliminate the time rate of change in density from the mass conservation equation as an unknown in the mass-momentum coupling. For this research effort, the energy coupling will be applied to the pressure Poisson equation through the equation of state, equation (1.24),

$$\frac{\partial P}{\partial t} = \left(\frac{\partial P}{\partial \rho} \right)_T \frac{\partial \rho}{\partial t} + \left(\frac{\partial P}{\partial T} \right)_\rho \frac{\partial T}{\partial t}. \quad (3.2)$$

The second term on the right-hand side of equation (3.2) is equivalent to the time rate of change in internal energy at constant density. By utilizing equation (1.11), this relationship can be expressed as

$$\left(\frac{\partial P}{\partial T} \right)_\rho \frac{\partial T}{\partial t} = \left(\frac{\partial P}{\partial e} \right)_\rho \frac{\partial e}{\partial t}. \quad (3.3)$$

The time rate of change in internal energy and temperature is related to the time rate of change in total energy by equations (1.10) and (1.12), respectively. For the PCICE algorithm, an explicit predictor for total energy is used to provide equation (3.2) with time-advanced internal energy information. Total energy is explicitly convected in a manner that is consistent with the explicit conservation of mass equation. The same explicit momentum components are used to convect both mass and total energy. The pressure Poisson equation thus has the energy information it requires to yield an accurate implicit pressure. A pressure-corrected total energy equation is derived that is also consistent with the pressure-correction equation for mass conservation. At the end of a time step, mass, momentum, and total energy are all pressure-corrected. As a result, the iterative process is not required which greatly reduces computer simulation time. This aspect is highly advantageous when computing transient simulations that are highly compressible and/or contains high energy deposition rates, chemical reactions, or phase change. This gain in efficiency is also apparent when marching to steady-state, which results in faster convergence.

The development of the PCICE-FEM scheme begins with the semi-implicit temporal discretization of the governing hydrodynamic equations. From these discretizations, the explicit predictor equations are derived. The balance of momentum equations are solved in fractional step form, similar to the form presented in Section 1.2, where the pressure gradient terms are implicit. The resulting explicit predictor equations for momentum are the time-weighted explicit momentum convective fluxes (in the absence of the pressure gradient terms), the components of the viscous stress tensor, and momentum source terms. In order to achieve second-order temporal accuracy, the momentum convective fluxes are cell-centered at

time $t = t^n + \varphi$, where φ is a time weighting parameter that varies between 0.0 and 1.0. When $\varphi = 0.5$, the momentum convective fluxes are exactly time-centered. For this study, the viscous stress tensor and momentum source terms are left at time $t = t^n$.

The semi-implicit mass conservation equation employs a time-weighted trapezoid rule for the mass flux temporal discretization. With the mass-momentum coupling procedure employed here, the semi-implicit balance of momentum equations are substituted for the implicit momentum term of the semi-implicit mass conservation flux. The resulting explicit components of this coupling constitute the explicit predictor for mass conservation. The explicit convective fluxes for mass conservation are time-weighted between the time-advanced explicit mass flux, obtained from the explicit balance of momentum equation, and the mass flux at time $t = t^n$.

The explicit predictor for the conservation of total energy equation is similar to the explicit predictor for mass conservation. The semi-implicit convective fluxes for total energy are also discretized with a time-weighted trapezoid rule. However, the implicit total enthalpy is a function of implicit pressure, implicit density, and explicit time-advanced total energy. The time-advanced total energy in the convective flux must be an explicit value because, as discussed before, eliminating implicit total energy in the convective flux term as an unknown is not possible. If total energy in the convective flux was defined implicit, the temporal discretization of the total energy equation would be fully implicit and would require an implicit solution algorithm. The explicit convective flux for total energy is obtained by multiplying the explicit mass conservation fluxes by the total enthalpy at time $t = t^n$. While the explicit convective flux for total energy can not be exactly time-centered, it will have the same convective velocity as the explicit mass conservation equation. Total energy will be exactly time-centered for $\varphi = 0.5$ after the pressure-correction equation for total energy is solved. The explicit conservation of total energy equation also includes the terms for viscous heating, heat conduction, and energy source, all at time $t = t^n$.

The mass-momentum coupling approach employed for the PCICE-FEM scheme results in a Poisson equation in terms of the change in pressure across a time step. This pressure Poisson equation is elliptic and requires a discretized algebraic system of equations to be solved simultaneously. The coefficient matrix derived from the spatial discretization of the governing partial differential equations is sparse. Also, the substitution of the equation of state for the partial derivative of density with respect to time renders the coefficient matrix to be non-symmetric. Direct methods, such as Gaussian elimination, are not applicable for solving large systems of equations. Direct methods require inversion and storage of the coefficient matrix and are extremely slow in actual computer time due to an excessively large amount of required computer operations. Furthermore, the discretization error is generally much larger than the error obtained with direct methods, which can approach machine error (the absolute number of significant digits the machine is capable of). There is, therefore, no reason then to solve the system of equations with a direct method. Efficient iterative methods that require several orders of magnitude less computational effort than direct methods have been developed for the type of problem presented here. The preconditioned Bi-Conjugate Gradient Stabilized (Bi-CGSTAB) iterative method, developed by H. A. Van Der Vorst (1992), was chosen for this research because of its ability to solve a non-symmetric system of equations

and for its smooth convergence. A new Jacobi preconditioner is developed here for Bi-CGSTAB based upon the coefficient matrix for the pressure Poisson equation. With this new diagonal preconditioner and exact finite element integration, no matrices are required by Bi-CGSTAB, only the assembly of matrix-vector dot products. Also, this new preconditioner has proven to be very efficient for Bi-CGSTAB, requiring only 4-7 iterations to achieve a reduction of four orders of magnitude in the relative error for change in pressure.

The final computational phase of the PCICE-FEM scheme is the semi-implicit pressure-correction of the conserved hydrodynamic variables. The pressure-correction equations are derived by subtracting the explicit predictor equations from the original semi-implicit temporal discretized governing equations. This includes the mass conservation equation. The original ICE scheme used the form of the equation of state that is substituted into the pressure Poisson equation to correct density. The correction equation for mass conservation in the PCICE-FEM scheme is composed of the explicit mass component and the pressure corrected divergence of momentum components. Using a discretized correction equation for mass conservation has shown to provide superior accuracy for the PCICE-FEM scheme over a correction equation derived from the equation of state when simulating compressible flows. Once the implicit pressure field is known, the implicit momentum components are determined. Next, the correction equation for mass conservation is solved with the convective mass flux determined from the implicit momentum solution to yield the implicit density field. And finally, the correction equation for total energy is solved. The convective flux for the total energy correction equation is composed of time-advanced implicit pressure, momentum, and density and the time-advanced explicit total energy from the explicit predictor phase. While these correction equations are semi-implicit in time, they are solved using the explicit iterative scheme, equation (2.17), described in Section 2.1.

The two-step Taylor-Galerkin FEM scheme, discussed in Chapter 2, is used to solve the explicit predictor equations for the balance of momentum in time and space. Both the explicit mass conservation and the explicit conservation of total energy equations are spatially discretized with the standard Galerkin FEM for unstructured meshes and solved with a time-weighted trapezoidal time integration scheme. It is at this point, and only this point, where artificial dissipation is applied to stabilize the solution near shocks and large changes in variable gradients for the PCICE-FEM scheme. After the explicit predictor equations have been solved, a high-resolution scheme or a variable diffusion is applied to control the numerical oscillations in the flow variables generated by the second-order spatial differencing. For the PCICE-FEM scheme, the high-resolution scheme is an improved version of the Flux-Corrected Transport (FCT) scheme described in Section 2.4.3 and the variable diffusion scheme is a modified form of Peraire's artificial dissipation found in Section 2.4.2. Adding artificial dissipation completes the explicit predictor phase of the PCICE-FEM scheme. After the explicit flow variables have been smoothed with artificial dissipation, the pressure Poisson equation is assembled from these smoothed components. Being elliptic in nature, the solution of the pressure Poisson equation creates no new extrema in the implicit pressure field. The implicit pressure field is then used to correct the explicit flow variables with the semi-implicit pressure-correction equations to yield the implicit flow variables. The point of not adding any artificial dissipation after the explicit predictor phase is an important one from a conservation standpoint. While the artificial dissipation schemes described here

are conservative, they are conservative in a global sense. Not adding artificial dissipation to the semi-implicit correction equations means that they are strictly conservative. For example, the correction equation for mass conservation yields a time rate of change in density at a node that is exactly equal (down to the discretization error) to the mass source plus the divergence of implicit momentum in the surrounding elements. This should improve the ability of PCICE-FEM scheme to simulate nearly incompressible flows.

This chapter contains a detailed development of the PCICE-FEM scheme. The semi-implicit temporal discretization of the PCICE algorithm will be presented first. From this temporal discretization, the derivations of the explicit predictor phase equations will then be presented. Following these derivations, the application of boundary conditions, the explicit solution procedure, and the application of artificial dissipation will be shown for the explicit predictor equations. The development and solution of the Poisson equation will then be presented. This will include derivation of the pressure Poisson equation from the semi-implicit temporal discretization of the governing equations, application of boundary conditions, and the solution by the iterative Bi-CGSTAB solver. The semi-implicit pressure-correction phase will then be covered and will include the derivation and solution of this last phase. The chapter will conclude with an algorithmic description and a discussion of the stability criteria for the PCICE-FEM scheme.

3.1 Temporal Discretization of the Governing Hydrodynamic Equations

The semi-implicit temporal discretization of the governing hydrodynamic equations for the PCICE algorithm is based upon the time-weighted trapezoid rule. In its basic form, the time-weighted trapezoid rule for equation (1.2) can be expressed as

$$U^{n+1} = U^n - \Delta t \vec{\nabla} \cdot \left[\phi \vec{F}^{n+1} + (1 - \phi) \vec{F}^n \right] + OT^n, \quad (3.4)$$

where U is the conserved variables, \vec{F} is the vector of convective fluxes, and OT^n denotes other terms, such as viscous and source terms. The superscripts n and $n+1$ denote values at the beginning and the end of a time step, respectively. The time-weighting parameter, ϕ , varies between 0.0 and 1.0. When $\phi = 0.5$, the convective fluxes are exactly time-centered. In the absence of OT^n , equation (3.4) is second-order accurate in time for $\phi = 0.5$. As before, Δt is the incremental time step size.

As discussed previously for a semi-implicit mass-momentum coupling procedure, only the momentum, pressure, and the density are treated implicitly. Therefore, the convective fluxes for the balance of momentum equations (except for pressure) are treated explicitly. An explicit approximation for the time-weighted convective fluxes is then required. Denoting the explicit balance of momentum convective fluxes by \vec{F}_m , the approximation of the time-weighted balance of momentum convective fluxes for the PCICE algorithm are

$$\vec{F}_m^{n+\phi} = F(U_e^{n+\phi}) \approx \phi \vec{F}_m^{n+1} + (1 - \phi) \vec{F}_m^n. \quad (3.5)$$

The superscript $n+\varphi$ denotes an intermediate time value at time $t = t^n + \varphi\Delta t$. For the PCICE-FEM scheme, the balance of momentum components of $U_e^{n+\varphi}$ are piecewise constant (cell-centered) value determined by a modified form of the first step of the two-step Taylor-Galerkin FEM scheme (see Section 3.2.4). The mass conservation and total energy components of $U^{n+\varphi}$ are nodal-based.

Employing equations (3.4) and (3.5), the semi-implicit temporal discretized governing equations, equation (1.2), are as follows:

for the conservation of mass equation,

$$\rho^{n+1} = \rho^n - \Delta t \vec{\nabla} \cdot \left[\varphi \rho \vec{u}^{n+1} + (1-\varphi) \rho \vec{u}^n \right] + \Delta t s^n, \quad (3.6)$$

for the balance of momentum equations,

$$\rho \vec{u}^{n+1} = \rho \vec{u}^n - \Delta t \vec{\nabla} \cdot (\rho \vec{u} \otimes \vec{u})^{n+\varphi} - \Delta t \vec{\nabla} \cdot \left[\varphi P^{n+1} + (1-\varphi) P^n \right] + \Delta t T_m^n + \Delta t \vec{d}^n, \quad (3.7)$$

and for the conservation of total energy equation,

$$\rho e_t^{n+1} = \rho e_t^n - \Delta t \vec{\nabla} \cdot \left[\varphi \rho \vec{u}^{n+1} H^{n+1} + (1-\varphi) \rho \vec{u}^n H^n \right] + \Delta t T_e^n + \Delta t i^n. \quad (3.8)$$

In equations (3.6) - (3.8), s , \vec{d} , and i are the mass, momentum, and total energy sources, respectively. T_m are the components of the viscous stress tensor and T_e is the viscous heat and heat conduction terms. The time-advanced total enthalpy in the total energy convective flux of equation (3.8) is defined by

$$H^{n+1} = \frac{\widetilde{\rho e_t} + P^{n+1}}{\rho^{n+1}}, \quad (3.9)$$

where $\widetilde{\rho e_t}$ is the explicit time-advanced total energy. The \sim is used to indicate that some form of artificial dissipation has been applied.

3.2 The Explicit Predictor Phase

The primary purpose of the explicit predictor phase of the PCICE algorithm is to provide the pressure Poisson equation with as much time-advanced information as possible. The explicit equations for the balance of momentum and the conservation of mass are derived from the mass-momentum coupling procedure. Substituting the semi-implicit balance of momentum equation, equation (3.7), into the implicit convective flux term of the semi-implicit mass conservation equation, equation (3.6), results in terms that are explicit in time. These explicit terms are collected to yield the explicit predictor equation for mass conservation. The explicit

predictor for the balance of momentum equations are necessary for three reasons. The first reason is that it is a much easier task to perform the divergence of an explicit time-advanced value of momentum in the explicit predictor equation for mass conservation than the divergence of the balance of momentum convective flux term, i.e., the second term on the right-hand side of equation (3.7). The second and third reasons have to do with the PCICE algorithm itself. The PCICE algorithm requires an explicit predictor value of time-advanced momentum for the trapezoidal time-weighting of the convective fluxes in the explicit predictor equations for mass conservation and conservation of total energy. And third, it is at the end of the explicit predictor phase where artificial dissipation is applied in the PCICE algorithm. The explicit total energy equation provides the pressure Poisson equation, through the equation of state, time-advanced change in internal energy information. It is designed to have the same convective velocities as the explicit predictor for mass conservation.

This section will first cover the derivation of the explicit predictor equations. Then the finite element spatial discretization for the explicit equations will be presented. The development of boundary conditions for the boundary integrals resulting from the finite element discretization will then be described. Next, the solution method for the discretized explicit equations will be presented. And finally, this section will end with a discussion of the application of artificial dissipation on the explicit solution.

3.2.1 Explicit Predictor Components of the Semi-Implicit Equations

Derivation of the explicit predictor phase equations will begin with the explicit predictor equation for the balance of momentum. The solution of this explicit equation is required for the trapezoid rule time-weighting of the convective fluxes in the explicit predictor equations for mass conservation and conservation of total energy. The derivation of the explicit predictor for the balance of momentum is elementary as the explicit predictor equation is composed of just the explicit portion of the semi-implicit balance of momentum equation, equation (3.7), without the gradient of pressure terms. This is the fractional step form described in Chorin (1967) and employed by Harlow and Amsden (1971) in the original ICE scheme. The form differs slightly here as the balance of momentum convective flux terms are explicitly time-weighted instead of being evaluated at time $t = t^n$. Collecting the explicit terms in equation (3.7), yields

$$\delta \rho \vec{u} = -\Delta t \vec{\nabla} \cdot (\rho \vec{u} \otimes \vec{u})^{n+\phi} + \Delta t T_m^n + \Delta t \vec{d}^n. \quad (3.10)$$

In equation (3.10), $\delta \rho \vec{u}$ is the explicit change in the momentum components for a given time step Δt defined by

$$\delta \rho \vec{u} = \rho \vec{u}^* - \rho \vec{u}^n, \quad (3.11)$$

where $\rho \vec{u}^*$ is the explicit time-advanced momentum solution. The * superscript indicates explicit time-advanced values without the application of artificial dissipation. The balance of

momentum flux in equation (3.10) at time $t = t^{n+\phi}$ are determined by a modified form of the first step of the two-step Taylor-Galerkin FEM. Equation (3.10) is similar to and has the same purpose of equation (1.22) in Section 1.2. In the Harlow and Amsden (1971) ICE formulation, the components of $\rho \vec{u}^*$ were defined by R and S , the explicit balance of momentum components in the r and z -directions, respectively.

The explicit predictor equation for mass conservation is derived from the mass-momentum coupling procedure. The explicit equation for mass conservation is found by collecting all of the explicit terms resulting from the substitution of the semi-implicit balance of momentum equation into the semi-implicit mass conservation equation. In order to simplify this procedure, equation (3.10) must first be solved. Substituting this result back into equation (3.7) yields

$$\rho \vec{u}^{n+1} = \rho \vec{u}^n + \delta \rho \vec{u} - \Delta t \vec{\nabla} \left[\phi P^{n+1} + (1 - \phi) P^n \right]. \quad (3.12)$$

Equation (3.12) is equation (3.7) with the explicit balance of momentum terms explicitly solved. Employing equation (3.12) in the mass-momentum coupling procedure instead of equation (3.7) alleviates the problem of performing the divergence of the time-weighted convective flux term found in equation (3.7). Substituting equation (3.12) into the implicit convective flux term on the right-hand side of equation (3.6) yields the basic form of the pressure Poisson equation,

$$\rho^{n+1} = \rho^n - \Delta t \vec{\nabla} \cdot \left(\phi \delta \rho \vec{u} + \rho \vec{u}^n \right) + \Delta t s^n + \phi \Delta t^2 \vec{\nabla} \cdot \vec{\nabla} \left[\phi P^{n+1} + (1 - \phi) P^n \right]. \quad (3.13)$$

The explicit predictor equation for mass conservation is found by collecting the explicit terms on the right-hand side of equation (3.13),

$$\delta \rho = -\Delta t \vec{\nabla} \cdot \left(\phi \delta \rho \vec{u} + \rho \vec{u}^n \right) + \Delta t s^n, \quad (3.14)$$

where $\delta \rho$ is the explicit change in mass for a given time step Δt . $\delta \rho$ is defined by

$$\delta \rho = \rho^* - \rho^n, \quad (3.15)$$

where ρ^* is the explicit time-advanced density.

For the PCICE algorithm, the explicit total energy equation provides the pressure Poisson equation, through the equation of state, time-advanced internal energy information. The explicit predictor equation for total energy is designed to be consistent with the explicit predictor for mass conservation. It employs the same trapezoid rule time-weighting for the convective velocities as the explicit predictor for mass conservation. The explicit convective flux for total energy is obtained by multiplying the mass conservation fluxes by the total enthalpy at time $t = t^n$. While the explicit convective flux for total energy can not be exactly

time-centered, it will have the same convective velocity as the explicit mass conservation equation. The explicit conservation of total energy equation also includes the terms for viscous heating, heat conduction, and energy source, all at time $t = t^n$. This explicit equation is written as

$$\delta \rho e_t = -\Delta t \vec{\nabla} \cdot \left[\left(\phi \delta \rho \vec{u} + \rho \vec{u}^n \right) H^n \right] + \Delta t T_e^n + \Delta t i^n, \quad (3.16)$$

where $\delta \rho e_t$ is the explicit change in total energy across a time step Δt . $\delta \rho e_t$ is defined by

$$\delta \rho e_t = \rho e_t^* - \rho e_t^n, \quad (3.17)$$

where ρe_t^* is the explicit time-advanced total energy.

3.2.2 Finite Element Discretization of the Explicit Predictor Equations

For the PCICE-FEM scheme, a modified form of the two-step Taylor-Galerkin FEM scheme (discussed in Chapter 2) is employed to solve the explicit predictor equations for the balance of momentum in time and space. The differences between the modified form and the original two-step form are that the convective fluxes contain no pressure terms and the modified two-step algorithm is evaluated around time $t^{n+\phi} = t^n + \phi \Delta t$ instead of $t^{n+\frac{1}{2}} = t^n + \frac{\Delta t}{2}$ as in the original form. The spatial discretization of the modified form is identical to the original form. Thus, a detailed finite element discretization of the explicit predictor for the balance of momentum equations is not presented here (see Section 2.1), only the final resulting system of finite element equations. The explicit predictor equations for mass conservation and total energy are temporally discretized with the time-weighted trapezoid rule found in equation (3.4). Spatial discretization is accomplished with the standard Galerkin FEM as the convective fluxes and dependent variables are located at the nodes of the elements. A detailed finite element discretization development for the explicit predictor equations for mass conservation and total energy is presented in Section 3.2.2.2.

3.2.2.1 Finite Element Discretization of the Explicit Predictor for the Balance of Momentum Equations

Considering the balance of momentum convective flux term of equation (3.10), it is obvious that both $\rho \vec{u}^{n+\phi}$ and $\rho^{n+\phi}$ will need to be determined in order to obtain $\vec{u}^{n+\phi}$. Thus, the approach for the PCICE-FEM scheme is to employ the first step of the two-step Taylor-Galerkin FEM scheme to solve the balance of momentum and mass conservation equations to obtain the values of $\rho \vec{u}^{n+\phi}$, $\rho^{n+\phi}$, and $\vec{u}^{n+\phi}$ for determining the balance of momentum convective flux term found in equation (3.10). With these variables known, equation (3.10) is solved with the same spatial discretization as the second step of the original two-step Taylor-Galerkin FEM scheme (see Section 2.1).

By employing the same finite element spatial discretizations as the first step of the original two-step Taylor-Galerkin FEM scheme, equation (2.6), the first step of the predictor phase for the balance of momentum equations is

$$\Omega_E U_E^{n+\varphi} = \int_{\Omega} [N] d\Omega \{U\}^n - \varphi \Delta t \int_{\Omega} [\vec{\nabla} N] d\Omega \cdot \{\vec{F}\}^n + \varphi \Delta t \Omega_E Q^n. \quad (3.18)$$

In equation (3.18), the components of $U_E^{n+\varphi}$ are

$$U_E^{n+\varphi} = \left\{ \begin{array}{c} \rho \\ \rho u \\ \rho v \end{array} \right\}_E^{n+\varphi} \quad (3.19)$$

and the vector components of \vec{F}^n in the x and y -directions are

$$f^n = \left\{ \begin{array}{c} \rho u \\ \rho u^2 \\ \rho uv \end{array} \right\}^n \quad \text{and} \quad g^n = \left\{ \begin{array}{c} \rho v \\ \rho uv \\ \rho v^2 \end{array} \right\}^n, \quad (3.20)$$

respectively.

The second solution step for the explicit balance of momentum predictor equations is spatially discretized in an identical fashion to the original two-step Taylor-Galerkin FEM scheme. The convective fluxes for equation (3.10) at time $t^{n+\varphi}$ are determined from the solution of equation (3.18). These cell-centered convective fluxes, denoted $\vec{F}_e^{n+\varphi}$, are substituted in place of $\vec{F}_e^{n+\frac{1}{2}}$ in equation (2.12) to yield the final form of the second step for the solution of the explicit balance of momentum predictor equations,

$$\begin{aligned} \sum_e \int_{\Omega} \{N\} [N] d\Omega \{\delta U\} &= \Delta t \sum_e \int_{\Omega} \{\vec{\nabla} N\} d\Omega \cdot \vec{F}_e^{n+\varphi} - \Delta t \sum_e \int_{\Gamma} \{N\} d\Gamma \vec{\eta} \cdot \vec{F}_e^{n+\varphi} \\ &\quad - \Delta t \sum_e \int_{\Omega} \{\vec{\nabla} N\} d\Omega \cdot \vec{F}_v^n + \Delta t \sum_e \int_{\Gamma} \{N\} d\Gamma \vec{\eta} \cdot \vec{F}_v^n + \Delta t \sum_e \int_{\Omega} \{N\} d\Omega Q_e^n. \end{aligned} \quad (3.21)$$

In equation (3.21), the components of δU , defined by equation (3.11), are

$$\delta U = \left\{ \begin{array}{c} \delta \rho u \\ \delta \rho v \end{array} \right\} = \left\{ \begin{array}{c} \rho u^* - \rho u^n \\ \rho v^* - \rho v^n \end{array} \right\}, \quad (3.22)$$

where the superscript * denotes time-advanced explicit values. The vector components of $\vec{F}_e^{n+\varphi}$ in the x and y -directions are

$$f_e^{n+\varphi} = \begin{Bmatrix} \rho u^2 \\ \rho uv \end{Bmatrix}_e^{n+\varphi} \quad \text{and} \quad g_e^{n+\varphi} = \begin{Bmatrix} \rho uv \\ \rho v^2 \end{Bmatrix}_e^{n+\varphi}, \quad (3.23)$$

respectively. The vector components of the viscous flux vector \vec{F}_v^n in the x and y -directions for the balance of momentum, defined in equation (1.1), are

$$f_v^n = \begin{Bmatrix} \tau_{xx} \\ \tau_{xy} \end{Bmatrix}^n \quad \text{and} \quad g_v^n = \begin{Bmatrix} \tau_{xy} \\ \tau_{yy} \end{Bmatrix}^n, \quad (3.24)$$

respectively. In equation (3.21), Q_e^n are the components of the balance of momentum sources found in equation (1.1). Evaluating the boundary integrals in equation (3.21) will be discussed later in Section 3.2.3.

3.2.2.2 Finite Element Discretization of the Explicit Predictor for the Mass Conservation and Total Energy Equations

For the PCICE-FEM scheme, both the explicit mass conservation and the total energy equations, equations (3.14) and (3.16), are spatially discretized with the standard Galerkin FEM. Both equations (3.14) and (3.16) are time integrated with an explicit form of the semi-implicit time-weighted trapezoid rule scheme found in equation (3.4). With viscous and source terms, the explicit form of equation (3.4) is expressed as

$$\delta U = -\Delta t \vec{\nabla} \cdot \vec{F}^* + \Delta t \vec{\nabla} \cdot \vec{F}_v^n + \Delta t Q^n. \quad (3.25)$$

In equation (3.25), δU is defined by equations (3.15) and (3.17) as

$$\delta U = \begin{Bmatrix} \delta \rho \\ \delta \rho e_t \end{Bmatrix} = \begin{Bmatrix} \rho^* - \rho^n \\ \rho e_t^* - \rho e_t^n \end{Bmatrix}, \quad (3.26)$$

where, as before, the superscript * denotes time-advanced explicit values. \vec{F}^* are the explicit time-advanced convective fluxes in equations (3.14) and (3.16),

$$\vec{F}^* = \begin{Bmatrix} \varphi \delta \rho \vec{u} + \rho \vec{u}^n \\ (\varphi \delta \rho \vec{u} + \rho \vec{u}^n) H^n \end{Bmatrix} = \begin{Bmatrix} \varphi \rho \vec{u}^* + (1 - \varphi) \rho \vec{u}^n \\ [\varphi \rho \vec{u}^* + (1 - \varphi) \rho \vec{u}^n] H^n \end{Bmatrix}. \quad (3.27)$$

In the x and y -directions, these convective fluxes are

$$f^* = \begin{Bmatrix} \varphi \delta \rho u + \rho u^n \\ (\varphi \delta \rho u + \rho u^n) H^n \end{Bmatrix} \text{ and } g^* = \begin{Bmatrix} \varphi \delta \rho v + \rho v^n \\ (\varphi \delta \rho v + \rho v^n) H^n \end{Bmatrix}, \quad (3.28)$$

respectively. The vector components of the viscous flux vector \vec{F}_v^n in the x and y -directions for mass conservation and total energy, defined in equation (1.1), are

$$f_v^n = \begin{Bmatrix} 0 \\ u \tau_{xx} + v \tau_{xy} - q_x \end{Bmatrix}^n \text{ and } g_v^n = \begin{Bmatrix} 0 \\ u \tau_{xy} + v \tau_{yy} - q_y \end{Bmatrix}^n, \quad (3.29)$$

respectively. In equation (3.25), Q^n are the components of the mass conservation and total energy sources found in equation (1.1).

The standard Galerkin finite element discretization of the explicit predictor equations for mass conservation and total energy begins with the method of weighted residuals. The weighted residual form of equation (3.25) is

$$\int_{\Omega} \delta U W d\Omega = -\Delta t \int_{\Omega} \vec{\nabla} \cdot \vec{F}^* W d\Omega + \Delta t \int_{\Omega} \vec{\nabla} \cdot \vec{F}_v^n W d\Omega + \Delta t \int_{\Omega} Q^n W d\Omega, \quad (3.30)$$

where W is some weight function with appropriate interpolation properties. Denoting the piecewise linear shape function associated with node j by N_j and the piecewise constant shape function associated with element e by P_e , the finite element approximations for equation (3.30) are

$$\delta U \approx \sum_j U_j N_j = [\delta U] \{N\}, \quad \vec{F}^* \approx \sum_j \vec{F}_j^* N_j = [\vec{F}^*] \{N\}, \quad \vec{F}_v^n \approx \sum_e \vec{F}_v^n P_e,$$

$$\text{and } Q^n = \sum_e Q_e^n P_e$$

where $P_e = 1$ on element e and $P_e = 0$ over the rest of the domain. The source terms Q_e^n and the viscous flux vector \vec{F}_v^n require piecewise constant representation because they are defined only on the elemental level. The sums over j and e are the sums over all the nodes and elements in the domain, respectively.

Applying the standard Galerkin approximation, $W = N_j$, and substituting the above finite element approximations into equation (3.30), yields

$$\begin{aligned} \sum_e \int_{\Omega} [\delta U] \{N\} N_j d\Omega = & -\Delta t \sum_e \int_{\Omega} \vec{\nabla} \cdot [\vec{F}^*] \{N\} N_j d\Omega + \Delta t \sum_e \int_{\Omega} \vec{\nabla} \cdot (\vec{F}_v^n P_e) N_j d\Omega \\ & + \Delta t \sum_e \int_{\Omega} Q_e^n P_e N_j d\Omega \end{aligned} \quad (3.31)$$

for each N_j . Integration by parts must be applied to the viscous flux integral on the right hand side of equation (3.31) in order to avoid the difficulty of performing the divergence of the viscous flux vector which is an elemental constant. This shifts part of the differentiability requirement from $\vec{F}_v^n P_e$ to N_j . Integration by parts may also be applied to the convective flux integral, even though the operation is not required to alleviate a problem of differentiability, in order to introduce flux boundary conditions. This shifts all of the differentiability requirement from $[\vec{F}] \{N\}$ to N_j .

Applying Green's theorem (integration by parts) and completing the matrix operations, equation (3.31) becomes

$$\begin{aligned} \sum_e \int_{\Omega} \{N\} [N] d\Omega \{\delta U\} = & \Delta t \sum_e \int_{\Omega} \{N\} [\vec{\nabla} N] d\Omega \cdot \{\vec{F}^*\} - \Delta t \sum_e \int_{\Gamma} \{N\} [N] d\Gamma \vec{\eta} \cdot \{\vec{F}^*\} \\ & - \Delta t \sum_e \int_{\Omega} P_e \{\vec{\nabla} N\} d\Omega \cdot \vec{F}_v^n + \Delta t \sum_e \int_{\Gamma} P_e \{N\} d\Gamma \vec{\eta} \cdot \vec{F}_v^n + \Delta t \sum_e \int_{\Omega} P_e \{N\} d\Omega Q_e^n, \end{aligned} \quad (3.32)$$

where Γ defines the boundary of the domain and $\vec{\eta}$ is the unit vector normal to the boundary. Evaluating the boundary integrals will be discussed later in Section 3.2.3.

Using the fact that P_e is nonzero only on element e , equation (3.32) becomes the final form of the discretized explicit predictor for the mass conservation and total energy equations,

$$\begin{aligned} \sum_e \int_{\Omega} \{N\} [N] d\Omega \{\delta U\} = & \Delta t \sum_e \int_{\Omega} \{N\} [\vec{\nabla} N] d\Omega \cdot \{\vec{F}^*\} - \Delta t \sum_e \int_{\Gamma} \{N\} [N] d\Gamma \vec{\eta} \cdot \{\vec{F}^*\} \\ & - \Delta t \sum_e \int_{\Omega} \{\vec{\nabla} N\} d\Omega \cdot \vec{F}_v^n + \Delta t \sum_e \int_{\Gamma} \{N\} d\Gamma \vec{\eta} \cdot \vec{F}_v^n + \Delta t \sum_e \int_{\Omega} \{N\} d\Omega Q_e^n. \end{aligned} \quad (3.33)$$

3.2.3 Boundary Conditions for the Explicit Predictor Equations

Equations (3.21) and (3.33) contain boundary integrals for the convective flux vector and the viscous stress tensor. The boundary integrals for the viscous stress tensor are evaluated in an identical fashion for the PCICE-FEM scheme as in the original two-step Taylor-Galerkin FEM scheme, described in Section 2.5.6. However, determination of the components of the convective flux vector for the boundary integrals in equations (3.21) and (3.33) differ considerably from the components found in the boundary integrals of equation (2.128). The first of two major differences is that the terms of the convective flux vector are at different time levels. And second, the balance of momentum equations for the explicit phase are in fractional step form. Thus, the pressure in the balance of momentum convective flux at the boundary is neglected. Referring to equation (2.48), the resulting components of the convective flux vector for the explicit predictor phase, without regard to time levels, are

$$f = \left\{ \begin{array}{c} m \\ \frac{m^2}{\rho} \\ \frac{mn}{\rho} \\ \frac{m(\varepsilon + P)}{\rho} \end{array} \right\} \quad g = \left\{ \begin{array}{c} n \\ \frac{mn}{\rho} \\ \frac{n^2}{\rho} \\ \frac{n(\varepsilon + P)}{\rho} \end{array} \right\}. \quad (3.34)$$

The flux Jacobians for this equation system are found by employing equations (2.51) and (2.54),

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -u^2 & 2u & 0 & 0 \\ -uv & v & u & 0 \\ -\gamma u e_t + (\gamma - 1) u \vec{u}^2 & \gamma e_t - \frac{\gamma - 1}{2} (3u^2 + v^2) & (1 - \gamma) uv & \gamma u \end{bmatrix} \quad (3.35)$$

and

$$B = \begin{bmatrix} 0 & 0 & 1 & 0 \\ -uv & v & u & 0 \\ -v^2 & 0 & 2v & 0 \\ -\gamma v e_t + (\gamma - 1) v \vec{u}^2 & (1 - \gamma) uv & \gamma e_t - \frac{\gamma - 1}{2} (u^2 + 3v^2) & \gamma v \end{bmatrix}. \quad (3.36)$$

The linear combination of A and B at the boundary can be defined by equation (2.56) in the unit outward normal direction,

$$K = \vec{A} \cdot \vec{\eta} = A\eta_x + B\eta_y. \quad (3.37)$$

The eigenvalues, or characteristic directions, of this system of equation is found by equation (2.57). The result is that

$$\lambda_1 = \lambda_2 = \lambda_3 = \lambda_4 = u_\eta. \quad (3.38)$$

With all of the eigenvalues repeated, there are important mathematical differences between the full system of governing hydrodynamic equations and the system of governing equations containing a fractional step form of the balance of momentum equations. Neglecting pressure from the balance of momentum equations has the effect of de-coupling the system of hydrodynamic equations for the explicit predictor phase. Without pressure in the balance of momentum equations, the equation of state no longer couples the mass conservation and total energy equations to the balance of momentum equations. This is reflected in the fact that there is no acoustic component in the eigenvalues of equation (3.38) as there is in equation (2.58). Thus, the wavelike solutions will consist only of waves propagating in the direction of material velocity. In fact, the system of equations can no longer be classified as strictly hyperbolic. The eigenvalues are real but the left eigenvectors are non-unique. Unlike the case presented in Section 2.5, all of the components of the left eigenvectors are arbitrary because all of the eigenvalues are repeated. Instead of one indeterminate, or arbitrary, component of the left eigenvectors due to $\lambda_1 = \lambda_2 = u_\eta$, as in equation (2.58), there are four indeterminate components. In other words, there is no linear independence of the left eigenvectors. Therefore, a characteristic form of the governing hydrodynamic equations with a fractional step form of the balance of momentum equations is not useful here for determining boundary conditions. Even though the characteristic boundary conditions are not implemented here, the characteristic analysis performed in Chapter 2 is useful. Because of the characteristics, it is known what can and can not be specified at the boundary for a given boundary condition type.

The procedure developed here for the PCICE-FEM scheme to approximate the terms of the convective flux vector on the boundary is to forward-time discretize the primitive variable form of the mass conservation and balance of momentum equations. To maintain consistency with the explicit predictor equations, the primitive variable form of the balance of momentum equations will be cast in fractional step form. A discretized energy equation at the boundary is not needed because the total enthalpy H term in equation (3.28) is at time $t = t^n$ and should

already be correct. Expanding equations (2.71) and (2.72), while neglecting the pressure gradient terms, yields

$$\frac{\partial \rho}{\partial t} + u \frac{\partial \rho}{\partial x} + v \frac{\partial \rho}{\partial y} + \rho \left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \right) = s, \quad (3.39)$$

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} = d_x, \quad (3.40)$$

and

$$\frac{\partial v}{\partial t} + u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} = d_y. \quad (3.41)$$

In equations (3.39) - (3.41) s , d_x , and d_y are primitive variable source terms.

Forward-time discretizing equations (3.39) - (3.41) to time $t = t^{n+\varphi}$ gives the change in density (ρ) and velocity (u and v) at the boundary,

$$\delta_b \rho = -\varphi \Delta t \left[s - \bar{u} \frac{\partial \rho^n}{\partial x} - \bar{v} \frac{\partial \rho^n}{\partial y} - \bar{\rho} \left(\frac{\partial u^n}{\partial x} + \frac{\partial v^n}{\partial y} \right) \right], \quad (3.42)$$

$$\delta_b u = \varphi \Delta t \left(d_x - \bar{u} \frac{\partial u^n}{\partial x} - \bar{v} \frac{\partial u^n}{\partial y} \right), \quad (3.43)$$

and

$$\delta_b v = \varphi \Delta t \left(d_y - \bar{u} \frac{\partial v^n}{\partial x} - \bar{v} \frac{\partial v^n}{\partial y} \right). \quad (3.44)$$

The variables with an overbar in equations (3.42) - (3.44) denote state variables on the boundary obtained by averaging the nodal variable values of a boundary face at time $t = t^n$. δ_b defines a change in the variable on the boundary face. These changes are added to the boundary node values at time $t = t^n$ to obtain boundary node values at the intermediate time level of $t = t^{n+\varphi}$. The derivatives and source terms are elemental values of the element adjacent to the boundary. For linear triangular finite elements, elemental derivatives and source terms are constant across the element, including the boundary. Equations (3.42) - (3.44) are discretized in a similar fashion to the first step of the adapted two-step Taylor-Galerkin FEM

scheme used for the balance of momentum explicit predictor, equation (3.18). The main differences being that equations (3.42) - (3.44) are in primitive variable form and that boundary values have been substituted where possible. The solution to equations (3.42) - (3.44) gives boundary node values of the convective flux vector terms for the balance of momentum equations, equation (3.23). The boundary solution also gives the opportunity to correct the momentum components on the boundary for the convective flux terms of mass conservation and total energy, equation (3.28).

3.2.4 Solution of the Explicit Predictor Equations

For the PCICE-FEM scheme, both the explicit predictor for the balance of momentums, equation (3.21), and the explicit predictors for mass conservation and total energy, equation (3.33), are cast in the form of equation (2.13) and iteratively solved with equation (2.17). Equation (3.21) is solved first to obtain $\rho \vec{u}^*$. Substituting this result into the explicit time-advanced convective fluxes of equation (3.27), equation (3.33) is solved to obtain ρ^* and ρe_t^* .

3.2.5 Application of Artificial Dissipation for the Components of the Explicit Predictor Phase of the PCICE-FEM scheme

The PCICE-FEM scheme is spatially second-order accurate and, therefore, is subject to high-order spatial effects for first-order derivatives which result in numerical oscillations of the flow variables in regions of large solution curvature. For the PCICE-FEM scheme, these high-order effects are generated primarily in the explicit predictor phase. Thus, the application of artificial dissipation for the PCICE algorithm only takes place in the explicit predictor phase. This is fortunate as it allows the artificial dissipation methods applied to the two-step Taylor-Galerkin FEM scheme to be used for the explicit predictor phase of the PCICE-FEM scheme. In fact, the artificial dissipation methods presented in Section 2.4, Lapidus artificial diffusion, Peraire's artificial dissipation, and Flux-Corrected Transport may be applied to the explicit predictor phase solution without modification. From this point on, the explicit time-advanced solution variables will be denoted as $\tilde{\rho}$, $\tilde{\rho \vec{u}}$, and $\tilde{\rho e_t}$ with the tilde \sim indicating that the solution variables from equations (3.22) and (3.26) have been smoothed with an artificial dissipation method. Upon smoothing,

$$\begin{aligned}\rho^* &\rightarrow \tilde{\rho} \\ \rho \vec{u}^* &\rightarrow \tilde{\rho \vec{u}} \\ \rho e_t^* &\rightarrow \tilde{\rho e_t}.\end{aligned}\tag{3.45}$$

In this section, the performance of these artificial dissipation methods with the PCICE-FEM scheme will be discussed. Also, improvements in these artificial dissipation methods have been developed in recent years and will be presented.

3.2.5.1 Lapidus Artificial Diffusion

The Lapidus artificial diffusion (Lapidus, 1967) method presented in Section 2.4.1 performed well with the PCICE-FEM scheme for simulations containing straight solid walls (see the shock tube simulation in Section 4.1.1). However for simulations containing curved solid walls, Lapidus artificial diffusion produced unacceptable undershoots of the flow variables in the vicinity of rapidly changing boundary geometry. Theoretically, the Lapidus artificial diffusion coefficient, k_l , should vanish at solid walls, either because the velocity is forced to zero for the no-slip solid wall boundary condition (viscous) or because the unit vector \vec{l} pointing in the direction of the maximum change in the absolute value of velocity for free-slip solid wall boundary condition (inviscid) is zero, $\vec{l} \cdot \vec{u} = 0$.

However, for a discretized triangular finite element domain, k_l does not completely vanish in the vicinity of curved boundaries because the derivatives of velocity are not calculated on the boundary. The derivatives are constants over a linear triangular finite element. Thus the values of velocity at the nodes opposite the boundary faces, where the no-slip or free-slip solid walls boundary conditions are not applied, may contribute to a non-zero elemental k_l . As discussed in Section 2.4.1, Löhner's formulation of Lapidus artificial diffusion (Löhner et al., 1985b) neglects the boundary integral resulting from integration by parts. Not applying the boundary integral in the Lapidus finite element formulation may result in an inconsistent formulation from a conservation point of view. Neglecting this boundary integral essentially sets $k_l = 0$ on the boundary when in fact k_l should not be zero due to integration along the boundary representing a curved surface. As a result, the simulated flow containing sudden changes in the boundary node normals (sudden change in boundary geometry) along an inviscid boundary experiences an undershoot in the velocity vector at the node opposite the boundary face toward the boundary. This undershoot tends to create a false boundary layer, even though the boundary is an inviscid surface. This undershoot effect is also apparent with the two-step Taylor-Galerkin FEM scheme but the magnitude is much smaller and seems to only have a localized effect, possibly due to the smaller time step restriction of the explicit scheme. None of this effect would be apparent in the original finite difference form of Lapidus artificial diffusion where integration by parts is not performed and k_l is a point form versus an elemental value for the finite element formulation.

3.2.5.2 Peraire's Artificial Dissipation

As discussed in Section 2.4.2, Peraire's finite element artificial dissipation method (Morgan and Peraire, 1987 and Peraire et al., 1988) was derived from a modified form of MacCormack and Baldwin's (1975) artificial viscosity. This dissipation method behaved satisfactorily with the PCICE-FEM scheme for all flow regimes. While this dissipation method is numerically inexpensive to apply, it tends to be more diffusive than a high-resolution method, such as Flux-Corrected Transport. Near contact discontinuities and slip lines, Peraire's artificial dissipation method is overly diffusive for transient simulations, even though pressure is constant across these flow phenomena and the resulting pressure sensor, equation (2.33), should be near zero. This overly diffusive effect is due to the linear nature of contact discontinuities and slip lines. With pressure waves propagating back and forth across the

domain in a transient simulation, Peraire's artificial dissipation method diffuses these flow features, and being linear phenomena, they tend to remain diffused. In steady-state applications, slip lines are recovered as the diffusion sensor approaches zero. Shock waves on the other hand are a nonlinear phenomena that tend toward a discontinuity in time. Physically, this is the result of intersecting flow characteristics (see Section 2.5 for a discussion on characteristic theory) on either side of the shock. However, Peraire's artificial dissipation method requires the diffusion coefficient, C_p in equation (2.31), to be set relatively high in order to control numerical oscillations in the vicinity of shocks.

An improvement to Peraire's artificial dissipation method has been developed by Swanson and Turkel (1992). This artificial dissipation method incorporates a weighted combination of Peraire's artificial dissipation method and Jameson's classic artificial viscosity method (Jameson et al., 1981). As with Peraire's artificial dissipation method, Jameson's artificial viscosity method also incorporates a diffusion coefficient based upon the second derivative of pressure. It was developed as a computationally inexpensive flow variable smoother for transonic inviscid flows. In its basic finite element form, the pressure sensor may be written as

$$S_p = \frac{h_e^2}{\bar{P}} \frac{\partial^2 P}{\partial x_j \partial x_j}, \quad (3.46)$$

where, as in equation (2.29), h_e is the element length and \bar{P} is the average elemental pressure. Combining equation (3.46) with equation (2.33) in a weighted fashion, the pressure sensor for the artificial dissipation method of Swanson and Turkel in finite element form (Löhner, 2001) is

$$S_p = \frac{\sum_e |2P_I - P_J - P_K|}{\sum_e [\alpha(|P_I - P_J| + |P_J - P_K|) + (1 - \alpha)\bar{P}_e]}, \quad (3.47)$$

where α is the weighting parameter and I, J, K refers to the local nodes numbers of element e . Typically, $\alpha = 0.5$ (Löhner, 2001).

For the PCICE-FEM scheme, the pressure sensor found in equation (3.47) is less sensitive in the vicinity of contact discontinuities and slip lines for transient simulations than the original sensor in equation (2.33). While still somewhat diffusive near shocks, this modified artificial dissipation method out-performs Peraire's original artificial dissipation method in the presence of shocks and is barely more expensive to compute. Surprisingly, this modified artificial dissipation method performed quite well with the PCICE-FEM scheme for nearly incompressible flows (see Chapter 4). For this type of flow, $\alpha = 0.95$ was found to provide the best results. With $\alpha = 0.95$, Swanson and Turkel's artificial dissipation method is weighted towards Peraire's artificial dissipation method. What is interesting is that Peraire's artificial dissipation method was designed for shock capturing in supersonic flows.

3.2.5.3 Flux-Corrected Transport

Löhner's (Löhner et al., 1987 and 1988) adaption of Zalesak's (1979) multi-dimensional Flux-Corrected Transport (FCT) scheme, discussed in Section 2.4.3, produced exceptional results for flows containing shock waves, especially for transient flow fields. With FCT applied to the PCICE-FEM scheme, shocks are captured within two to three computational cells. Numerical oscillations are virtually eliminated with the application of FCT, even for the strongest shocks (see the laboratory accident simulation in Section 4.1.2). However, there is an inherent problem with FCT in regions of flow defined by an expansion fan (rarefaction wave). In these specific regions of the flow, the FCT solution can result in a stair-stepping profile of the flow variables. While still monotonic, the non-physical stair-stepping phenomena nonetheless produces an inaccurate representation of the flow field. While Boris and Book (1973, 1975, and 1976) noted this problem with the original FCT scheme and Zalesak's (1979) multi-dimensional FCT scheme reduced the magnitude of this problem, the problem still exists in Löhner's finite element formulation of FCT. Löhner does not describe this problem (or solution) in any of his journal articles and he only alludes to it in his new text (Löhner, 2001). Löhner's (1994) solution to this stair-stepping problem is to apply a very small amount of artificial dissipation after the application of FCT. For the PCICE-FEM scheme this is accomplished by setting $C_k = 0.05-0.1$ in equation (2.22) or $C_p = 0.02-0.05$ in equation (2.32), depending on which artificial dissipation method is employed. These values of the diffusion coefficients were found empirically and are somewhat problem dependent.

An improvement to Löhner's Finite Element Flux-Corrected Transport (FEM-FCT) scheme for the solution of hydrodynamic conservation equations was developed by Georghiou et al. (1999). Georghiou uses the same high and low-order schemes that were developed by Löhner (1987), equations (2.13) and (2.39), respectively. The original low-order scheme is obtained by adding diffusion to the lumped-mass form of the high-order scheme by subtracting the lumped-mass matrix from the consistent-mass matrix, equation (2.38). Whereas Löhner employed a globally constant diffusion coefficient, $C_d = 1.0$, Georghiou's modification employs a globally varying diffusion coefficient. This new diffusion coefficient is assumed to be constant in each element and renders the low-order scheme equivalent to an upwind scheme which has the minimum amount of diffusion required to ensure monotonic results. The form of the new elemental diffusion coefficient is

$$C_d = \frac{C_e(1.0 - C_e)}{2}, \quad (3.48)$$

where C_e is the elemental material Courant number. C_e is defined by

$$C_e = \frac{|\vec{u}_e| \Delta t}{h_e}, \quad (3.49)$$

where $|\vec{u}_e|$ is the magnitude of the elemental velocity found by averaging the nodal values of velocity in element e .

By itself, the performance of the new low-order scheme incorporated into the PCICE-FEM scheme approaches the quality of results obtained by Peraire's original artificial dissipation method in the vicinity of shocks and expansion fans. However, it suffers in resolving contact discontinuities and slip lines by nearly diffusing them out of the flow field. When combined with the high-order scheme, the new low-order scheme provides superior FCT results (see the shock tube simulation in Section 4.1.1) with a minimal increase in computational cost.

3.3 The Elliptic Pressure Poisson Equation

One of the main objectives of this research is to couple the total energy equation to the mass-momentum coupled pressure Poisson equation in a strong enough manner to avoid the iterative procedure with the governing hydrodynamic equations (see Section 1.2.1). This is accomplished in the PCICE algorithm by performing an operator splitting of the time-discretized terms of the total energy equation into an explicit predictor and a semi-implicit corrector. The information obtained from the explicit total energy solution is used to satisfy the time-advanced energy requirements of the equation of state, the second term on the right-hand side of equation (3.2). The equation of state is used to eliminate the implicit density as an unknown and fully express the mass conservation equation in terms of the implicit pressure. With the time-advanced solution from the pressure Poisson equation, the semi-implicit correction for total energy may then be obtained.

This section is devoted to describing all aspects of the derivation, finite element spatial discretization, and solution of the pressure Poisson equation incorporated into the PCICE-FEM scheme. The pressure Poisson equation will be derived with the semi-implicit time-discretized governing hydrodynamic equations of Section 3.1. A complete development of the spatial discretization with the standard Galerkin finite element method will then be presented. Boundary conditions for the pressure Poisson equation that are consistent with the finite element spatial discretization and the physics of the governing hydrodynamic equations will then be developed. This section will conclude with a description of the preconditioned Bi-Conjugate Gradient Stabilized iterative method of solving the pressure Poisson equation for the PCICE-FEM scheme.

3.3.1 Derivation of the Pressure Poisson Equation

The basic mass-momentum coupled pressure-based algorithm, presented in Section 1.2.1, incorporates a pressure Poisson equation that is derived by substituting the balance of momentum equations into the mass conservation equation. This mass-momentum coupled algorithm requires pressure, momentum, and density to be treated implicitly while the total energy is solved in an explicit fashion. Similarly, the basic energy-momentum coupled pressure-based algorithm presented in Section 1.2.2, employs a pressure Poisson equation that is derived by substituting the balance of momentum equations into the total energy equation. This energy-momentum algorithm requires that pressure, momentum, and total energy be treated in an implicit fashion while the mass conservation equation is solved for explicitly. However, in order for total energy to be treated in a fully implicit manner, the total energy

variable in the convective flux term of equation (1.20) must also be a time-advanced implicit value. While the implicit momentum variable may be eliminated from the total energy convective flux function by direct substitution of the semi-implicit balance of momentum equations, there is no apparent method of eliminating an implicit total energy variable from this convective flux term. The total energy variable in the convective flux term must either be left at old-time or, as with the PCICE algorithm, explicitly predicted. Therefore, the total energy variable for an energy-momentum coupled algorithm is actually treated in a semi-implicit manner. The PCICE algorithm is a mass-momentum coupled pressure-based algorithm that combines the level of implicitness from both the mass-momentum coupled and energy-momentum coupled algorithms. Due to the mass-momentum coupling procedure, the PCICE algorithm requires pressure, momentum, and density to be implicit. However, instead of the total energy being treated as an explicit quantity, total energy is solved in a semi-implicit manner. The PCICE algorithm solves the same semi-implicit total energy equation that would be required for a time-weighted energy-momentum coupled algorithm.

To begin the mass-momentum coupled derivation for the PCICE algorithm, the time-weighted semi-implicit balance of momentum equations, equation (3.7), is rewritten in terms of the explicit change in the balance of momentum components and implicit pressure,

$$\rho \vec{u}^{n+1} = \rho \vec{u}^n + \delta \widetilde{\rho \vec{u}} - \Delta t \vec{\nabla} [\phi P^{n+1} + (1 - \phi) P^n] . \quad (3.50)$$

In equation (3.50), $\delta \widetilde{\rho \vec{u}}$ is defined as the explicit change in the balance of momentum over a time step Δt ,

$$\delta \widetilde{\rho \vec{u}} = \widetilde{\rho \vec{u}} - \rho \vec{u}^n . \quad (3.51)$$

$\delta \widetilde{\rho \vec{u}}$ is obtained by solving equation (3.10) with an appropriate amount of artificial dissipation to control numerical oscillations. As will be discussed in Section 3.4, equation (3.50) is the semi-implicit pressure correction equation for the balance of momentum.

Substituting equation (3.50) into the semi-implicit mass conservation equation, equation (3.6), to eliminate the implicit balance of momentum components $\rho \vec{u}^{n+1}$ as unknowns, yields the basic form of the pressure Poisson equation,

$$\rho^{n+1} - \rho^n = -\Delta t \vec{\nabla} \cdot (\phi \delta \widetilde{\rho \vec{u}} + \rho \vec{u}^n) + \Delta t s^n + \phi \Delta t^2 \vec{\nabla} \cdot \vec{\nabla} [\phi P^{n+1} + (1 - \phi) P^n] . \quad (3.52)$$

Equation (3.52) is the semi-implicit mass conservation equation in terms of the explicit mass conservation flux and source terms, the implicit time-weighted pressure, and the implicit change in density. At this point, a decision must be made whether to express the pressure Poisson equation in terms of the implicit time-advanced pressure or an implicit pressure change variable. For the PCICE algorithm, the pressure change form was chosen for ease in

applying Dirichlet boundary conditions in the iterative solver (see Section 3.3.4). Introducing the implicit pressure change variable as

$$\delta P = \varphi(P^{n+1} - P^n) , \quad (3.53)$$

equation (3.52) is then recast into the form of

$$\rho^{n+1} - \rho^n = -\Delta t \vec{\nabla} \cdot (\varphi \delta \widetilde{\rho \vec{u}} + \rho \vec{u}^n) + \Delta t s^n + \varphi \Delta t^2 \vec{\nabla} \cdot \vec{\nabla} (\delta P + P^n) . \quad (3.54)$$

Equation (3.54) contains two unknowns, ρ^{n+1} and δP . Because δP is the desired dependent variable from which to form the Poisson equation, the equation of state is used to express the left-hand side of equation (3.54) in terms of δP . Time discretizing the equation of state, equation (3.2), yields

$$\frac{P^{n+1} - P^n}{\Delta t} = \left(\frac{\partial P}{\partial \rho} \right)_T \frac{\rho^{n+1} - \rho^n}{\Delta t} + \left(\frac{\partial P}{\partial T} \right)_\rho \frac{T^{n+1} - T}{\Delta t} , \quad (3.55)$$

where, for an ideal gas,

$$\left(\frac{\partial P}{\partial \rho} \right)_T = R_c \bar{T} \quad (3.56)$$

and

$$\left(\frac{\partial P}{\partial T} \right)_\rho = R_c \bar{\rho} . \quad (3.57)$$

The overbar in equations (3.56) and (3.57) denotes state variables. In a time-discretized sense, the state variables may be at any time level between $t = t^n$ and $t = t^{n+1}$. Thus, the state variables are defined in the time-weighted manner consistent with the PCICE algorithm,

$$\bar{T} = \varphi T^{n+1} + (1 - \varphi) T^n \quad (3.58)$$

and

$$\bar{\rho} = \varphi \rho^{n+1} + (1 - \varphi) \rho^n . \quad (3.59)$$

With the substitution of equations (3.56) and (3.57) in terms of equations (3.58) and (3.59), the time-discretized equation of state is now

$$\frac{P^{n+1} - P^n}{\Delta t} = R_c \bar{T} \frac{\rho^{n+1} - \rho^n}{\Delta t} + R_c \bar{\rho} \frac{T^{n+1} - T^n}{\Delta t}. \quad (3.60)$$

Equation (3.60) is now expressed in terms of the pressure Poisson equation's dependent variable, δP , as

$$\frac{\delta P}{\phi} = R_c \bar{T} (\rho^{n+1} - \rho^n) + R_c \bar{\rho} (T^{n+1} - T^n). \quad (3.61)$$

Solving equation (3.61) for the change in density across a time step,

$$\rho^{n+1} - \rho^n = \frac{\delta P}{\phi R_c \bar{T}} - \frac{\bar{\rho}}{\bar{T}} (T^{n+1} - T^n), \quad (3.62)$$

and substituting into left-hand side of equation (3.54) to eliminate the implicit density ρ^{n+1} as an unknown yields

$$\begin{aligned} \frac{\delta P}{\phi R_c \bar{T}} - \phi \Delta t^2 \vec{\nabla} \cdot \vec{\nabla} \delta P &= \frac{\bar{\rho}}{\bar{T}} (T^{n+1} - T^n) - \Delta t \vec{\nabla} \cdot (\phi \delta \widetilde{\rho \vec{u}} + \rho \vec{u}^n) \\ &\quad + \Delta t s^n + \phi \Delta t^2 \vec{\nabla} \cdot \vec{\nabla} P^n. \end{aligned} \quad (3.63)$$

The substitution of the equation of state for the time rate of change in density term of the mass conservation equation is designed to eliminate the implicit density as an unknown in the pressure Poisson equation. For Harlow and Amsden's (1971) original ICE scheme, the effective barotropic equation of state, equation (3.1), which neglected the implicit dependence of pressure on internal energy, was substituted to eliminate the implicit density. Harlow and Amsden also left the temperature state variable at old-time, $t = t^n$. Therefore, no new-time variables were introduced into the pressure Poisson equation with the equation of state substitution. However, the full equation of state, equation (3.2), incorporated into the pressure Poisson equation introduces a dependence on internal energy expressed in the new-time temperature, T^{n+1} . The first term on the right-hand side of equation (3.63) is the coupling of the pressure Poisson equation to internal energy and thus to the total energy equation. If the internal energy term of equation (3.63) is to be fully addressed at time $t = t^{n+1}$, then an iterative procedure between the pressure Poisson equation and the governing hydrodynamic equations will have to be implemented.

On the other hand, if the variables of the internal energy term of equation (3.63) can be sufficiently predicted, the iterative procedure can be avoided. The PCICE algorithm

accurately predicts the new-time variables of the internal energy time by employing the explicit predictor solution of the hydrodynamic variables, equation (3.45). A new-time temperature field, \tilde{T} , is determined from the explicit predictor solution with the equation of state, equation (1.12). The time-advanced terms of the state variables are also determined from the explicit predictor solution. With substitution of these predicted variables into equation (3.63), the final form of the pressure Poisson equation for the PCICE algorithm becomes

$$\begin{aligned} \frac{\delta P}{\phi R_c \bar{T}} - \phi \Delta t^2 \vec{\nabla} \cdot \vec{\nabla} \delta P = \frac{\bar{\rho}}{\bar{T}} (\tilde{T} - T^n) - \Delta t \vec{\nabla} \cdot (\phi \delta \widetilde{\rho \vec{u}} + \rho \vec{u}^n) \\ + \Delta t s^n + \phi \Delta t^2 \vec{\nabla} \cdot \vec{\nabla} P^n, \end{aligned} \quad (3.64)$$

with the state variables determined by

$$\bar{T} = \phi \tilde{T} + (1 - \phi) T^n \quad (3.65)$$

and

$$\bar{\rho} = \phi \tilde{\rho} + (1 - \phi) \rho^n. \quad (3.66)$$

Equation (3.64) is an elliptic equation in terms of the one unknown dependent variable, δP , on the left-hand side and the right-hand side is constructed of explicit terms only and, therefore, there is no need to iterate with the semi-implicit governing equations. This equation is actually a mixed elliptic-hyperbolic equation. It derives its elliptic nature from the Laplacian operators on δP and P^n . Without these operators, the equation represents an explicit mass conservation equation in conservative form, which is hyperbolic in nature. For low-speed flows approaching the incompressible limit, the Laplacian terms tend to dominate as density becomes uniform and the divergence of velocity approaches zero. For high-speed flows where the time step Δt becomes small, the convective mass flux terms dominate as the coefficient of the Laplacian terms $\phi \Delta t^2$ becomes very small.

3.3.2 Finite Element Discretization of the Pressure Poisson Equation

For the PCICE-FEM scheme, the pressure Poisson equation is spatially discretized by the standard Galerkin finite element method. Beginning with the weighted residual form of equation (3.64)

$$\begin{aligned} \int_{\Omega} \frac{\delta P}{\phi R_c \bar{T}} W d\Omega - \phi \Delta t^2 \int_{\Omega} \vec{\nabla} \cdot \vec{\nabla} \delta P W d\Omega = \int_{\Omega} \frac{\bar{\rho}}{\bar{T}} (\tilde{T} - T^n) W d\Omega \\ - \Delta t \int_{\Omega} \vec{\nabla} \cdot (\phi \delta \widetilde{\rho \vec{u}} + \rho \vec{u}^n) W d\Omega + \Delta t \int_{\Omega} s^n W d\Omega \\ + \phi \Delta t^2 \int_{\Omega} \vec{\nabla} \cdot \vec{\nabla} P^n W d\Omega, \end{aligned} \quad (3.67)$$

where W is some weight function with appropriate interpolation properties. Denoting the piecewise linear shape function associated with node j by N_j and the piecewise constant shape function associated with element e by P_e , the finite element approximations to be used in equation (3.67) are

$$\begin{aligned}\delta P &\approx \sum_j \delta P_j N_j = [\delta P]\{N\}, \quad P^n \approx \sum_j P_j^n N_j = [P^n]\{N\}, \\ \delta \widetilde{\rho \vec{u}} &\approx \sum_j \delta \widetilde{\rho \vec{u}}_j N_j = [\delta \widetilde{\rho \vec{u}}]\{N\}, \quad \rho \vec{u}^n \approx \sum_j \rho \vec{u}_j^n N_j = [\rho \vec{u}^n]\{N\}, \\ \frac{\bar{\rho}}{\bar{T}}(\tilde{T} - T^n) &\approx \sum_j \frac{\bar{\rho}}{\bar{T}}(\tilde{T} - T^n)_j N_j = \left[\frac{\bar{\rho}}{\bar{T}}(\tilde{T} - T^n) \right]\{N\}, \quad s^n = \sum_e s_e^n P_e,\end{aligned}\tag{3.68}$$

where $P_e = 1$ on element e and $P_e = 0$ over the rest of the domain. s_e^n requires piecewise constant representation because the source term is defined only on the elemental level. The sums over j and e are the sums over all the nodes and elements in the domain, respectively.

Applying the standard Galerkin approximation, $W = N_j$, and substituting the above finite element approximations into equation (3.67), yields

$$\begin{aligned}\sum_e \int_{\Omega} \left[\frac{1}{\phi R_c \bar{T}} \right] [\delta P]\{N\} N_j d\Omega - \phi \Delta t^2 \sum_e \int_{\Omega} \vec{\nabla} \cdot \vec{\nabla} [\delta P]\{N\} N_j d\Omega = \\ \sum_e \int_{\Omega} \left[\frac{\bar{\rho}}{\bar{T}}(\tilde{T} - T^n) \right] [N] N_j d\Omega - \Delta t \sum_e \int_{\Omega} \vec{\nabla} \cdot [\phi \delta \widetilde{\rho \vec{u}} + \rho \vec{u}^n]\{N\} N_j d\Omega \\ + \Delta t \sum_e \int_{\Omega} s_e^n P_e N_j d\Omega + \phi \Delta t^2 \sum_e \int_{\Omega} \vec{\nabla} \cdot \vec{\nabla} [P^n]\{N\} N_j d\Omega,\end{aligned}\tag{3.69}$$

for each N_j . The first term of equation (3.69) contains a diagonal coefficient matrix resulting from the substitution of the equation of state for the time rate of change in density. This matrix is defined by

$$\left\{ \frac{\delta P}{\phi R_c \bar{T}} \right\} = \left[\frac{1}{\phi R_c \bar{T}} \right] \{ \delta P \} .\tag{3.70}$$

Integration by parts must be applied to the integrals of equation (3.69) containing a Laplacian operator on pressure because of the differentiability constraints imposed by linear shape functions. Integration by parts shifts part of the differentiability requirement from $[\delta P]\{N\}$ and $[P^n]\{N\}$ to N_j . The result of this operation is that the Laplacian operator is transformed from a second-order differential to the product of two first-order differentials. Integration by

parts is also applied to the explicit time-weighted convective mass flux integral. Even though the operation is not required to alleviate a problem of differentiability, the operation is performed in order to introduce mass flux boundary conditions. This shifts all of the differentiability requirement from $[\varphi\delta\widetilde{\rho\vec{u}} + \rho\vec{u}^n]\{N\}$ to N_j . Applying Green's theorem (integration by parts) and completing the matrix operations, equation (3.69) becomes

$$\begin{aligned}
& \sum_e \int_{\Omega} \{N\}[N]d\Omega \left[\frac{1}{\varphi R_c \bar{T}} \right] \{\delta P\} + \varphi \Delta t^2 \sum_e \int_{\Omega} \{\vec{\nabla} N\} \cdot [\vec{\nabla} N] d\Omega \{\delta P\} \\
& - \varphi \Delta t^2 \sum_e \int_{\Gamma} \{N\} d\Gamma \left\{ \frac{\partial \delta P}{\partial \vec{\eta}} \right\} = \sum_e \int_{\Omega} \{N\}[N]d\Omega \left\{ \frac{\bar{\rho}}{\bar{T}} (\tilde{T} - T^n) \right\} \\
& + \Delta t \sum_e \int_{\Omega} \{\vec{\nabla} N\}[N]d\Omega \cdot \{\widetilde{\rho\vec{u}}^{n+\varphi}\} - \Delta t \sum_e \int_{\Gamma} \{N\}[N]d\Gamma \{\widetilde{\rho\vec{u}}_{\eta}^{n+\varphi}\} \\
& + \Delta t \sum_e \int_{\Omega} \{N\}d\Omega s_e^n - \varphi \Delta t^2 \sum_e \int_{\Omega} \{\vec{\nabla} N\} \cdot [\vec{\nabla} N] d\Omega \{P^n\} \\
& + \varphi \Delta t^2 \sum_e \int_{\Gamma} \{N\} d\Gamma \left\{ \frac{\partial P^n}{\partial \vec{\eta}} \right\}, \tag{3.71}
\end{aligned}$$

where, as before, Γ defines the boundary of the domain and $\vec{\eta}$ is the unit vector normal to the boundary. In the convective mass flux volume integral of equation (3.71), the explicit time-weighted convective mass flux has been redefined as

$$\widetilde{\rho\vec{u}}^{n+\varphi} = \varphi \delta \widetilde{\rho\vec{u}} + \rho\vec{u}^n \tag{3.72}$$

and the normal component of this variable found in the convective mass flux boundary integral is defined as

$$\widetilde{\rho\vec{u}}_{\eta}^{n+\varphi} = \vec{\eta} \cdot \widetilde{\rho\vec{u}}^{n+\varphi}. \tag{3.73}$$

The normal pressure gradients contained in the boundary integrals of equation (3.71) are expressed as

$$\left\{ \frac{\partial \delta P}{\partial \vec{\eta}} \right\} = \vec{\eta} \cdot [\vec{\nabla} N] \{\delta P\}$$

and

$$\left\{ \frac{\partial P^n}{\partial \vec{\eta}} \right\} = \vec{\eta} \cdot [\vec{\nabla} N] \{P^n\}.$$

3.3.3 Boundary Conditions for the Pressure Poisson Equation

The spatial integration of the pressure Poisson equation derived for the PCICE-FEM scheme has resulted in a set of boundary conditions that are not typically found in the governing hydrodynamic equations describing a fluid continuum. Equation (3.71) contains boundary integrals requiring knowledge of the pressure gradients normal to the boundary for both the time-advanced pressure P^{n+1} , in terms of δP , and the old-time pressure P^n . These normal pressure gradients, commonly referred to as Neumann boundary conditions, are generally unknown. Not only are they unknown, the characteristic analysis of the governing hydrodynamic equations for compressible flow, performed in Section 2.5, shows that the normal pressure gradients on a boundary are never to be specified. What the characteristic analysis does show is that either the pressure or the velocity (but not both) must be specified for subsonic flow boundaries. While an analysis of the governing hydrodynamic equations may determine what the normal pressure gradients should be near an inflow or outflow boundary, the application of normal pressure gradient boundary conditions are not a requirement to achieve a unique solution and may be considered to be redundant. At a stationary solid wall where the mass flux is specified to be zero, it can be effectively argued that specification of normal pressure gradient boundary conditions over-specifies the boundary. Therefore, the correct boundary conditions for the pressure Poisson equation lie in the application of Dirichlet boundary conditions for δP or the specification of momentum in the normal direction as a Neumann boundary in the mass flux boundary integrals of equation (3.71).

Gresho and Sani (1987 and 1999) give an extensive discussion of the problems created with the introduction of the normal pressure gradient boundary conditions (Neumann) for incompressible flows. They state that the Neumann boundary condition is always appropriate for the pressure Poisson equation, for both the set of initial conditions and at solution times greater than zero. However, what they call a Neumann boundary condition is the result of replacing the normal pressure gradient at the boundary with the normal momentum equation applied at the boundary. Thus, all of the boundary conditions in the forcing function of the pressure Poisson equation are in terms of velocity. This is precisely the approach taken here for the PCICE-FEM scheme. Veldman (1990) summarized this approach and applied it to other systems of governing equations, namely the shallow water equations and the alternating direction implicit (ADI) scheme for Poisson equations. Veldman described the approach as “first discretize the equations of motion, next substitute the original boundary conditions, and finally combine the discrete equations.”

The first step in applying Gresho and Sani’s approach to constructing an appropriate set Neumann boundary conditions for the pressure Poisson equation of the PCICE-FEM scheme

is to write the semi-implicit balance of momentum equations in terms of the boundary variables found in equation (3.71). Multiplying equation (3.50) by φ and expressing the time-weighted pressure gradient in terms of δP gives

$$\varphi \rho \vec{u}^{n+1} - \varphi \rho \vec{u}^n = \varphi \delta \widetilde{\rho \vec{u}} - \varphi \Delta t \vec{\nabla}(\delta P + P^n). \quad (3.74)$$

Now add $\rho \vec{u}^n$ to both sides of equation (3.74) to give

$$\varphi \rho \vec{u}^{n+1} + (1 - \varphi) \rho \vec{u}^n = \varphi \delta \widetilde{\rho \vec{u}} + \rho \vec{u}^n - \varphi \Delta t \vec{\nabla}(\delta P + P^n). \quad (3.75)$$

Making use of equation (3.72), equation (3.75) is expressed as

$$\rho \vec{u}^{n+\varphi} = \widetilde{\rho \vec{u}}^{n+\varphi} - \varphi \Delta t \vec{\nabla}(\delta P + P^n), \quad (3.76)$$

where the left-hand side of equation (3.75) is written in terms of the implicit time-weighted momentum variable defined as

$$\rho \vec{u}^{n+\varphi} = \varphi \rho \vec{u}^{n+1} + (1 - \varphi) \rho \vec{u}^n. \quad (3.77)$$

Equation (3.76) is now multiplied by Δt and discretized in a finite element manner along the domain boundary Γ to yield

$$\begin{aligned} \Delta t \int_{\Gamma} \{N\} [N] d\Gamma \{ \rho u_{\eta}^{n+\varphi} \} &= \Delta t \int_{\Gamma} \{N\} [N] d\Gamma \{ \widetilde{\rho u_{\eta}}^{n+\varphi} \} \\ &\quad - \varphi \Delta t^2 \int_{\Gamma} \{N\} d\Gamma \left\{ \frac{\partial(\delta P + P^n)}{\partial \vec{\eta}} \right\}. \end{aligned} \quad (3.78)$$

Solving equation (3.78) for the normal pressure gradient boundary integrals and substituting into equation (3.71) yields

$$\begin{aligned} \sum_e \int_{\Omega} \{N\} [N] d\Omega \left[\frac{1}{\varphi R_c \bar{T}} \right] \{ \delta P \} &+ \varphi \Delta t^2 \sum_e \int_{\Omega} \{ \vec{\nabla} N \} \cdot [\vec{\nabla} N] d\Omega \{ \delta P \} = \\ \sum_e \int_{\Omega} \{N\} [N] d\Omega \left\{ \frac{\bar{\rho}}{\bar{T}} (\tilde{T} - T^n) \right\} &+ \Delta t \sum_e \int_{\Omega} \{ \vec{\nabla} N \} [N] d\Omega \cdot \{ \widetilde{\rho \vec{u}}^{n+\varphi} \} \\ - \Delta t \sum_e \int_{\Gamma} \{N\} [N] d\Gamma \{ \rho u_{\eta}^{n+\varphi} \} &+ \Delta t \sum_e \int_{\Omega} \{N\} d\Omega s_e^n \\ - \varphi \Delta t^2 \sum_e \int_{\Omega} \{ \vec{\nabla} N \} \cdot [\vec{\nabla} N] d\Omega \{ P^n \}, \end{aligned} \quad (3.79)$$

where the normal component of the implicit time-weighted mass flux variable is defined as

$$\rho \vec{u}_\eta^{n+\varphi} = \vec{\eta} \cdot \rho \vec{u}^{n+\varphi} . \quad (3.80)$$

Equation (3.79) is the final form of the discretized pressure Poisson equation for the PCICE-FEM scheme. Note that the normal pressure gradient boundary integrals have been eliminated. The Neumann boundary condition is now satisfied through the convective mass flux boundary integral. Dirichlet boundary conditions are applied through the specification of δP . Therefore, along the boundary, either the normal component of the implicit time-weighted mass flux, $\rho \vec{u}_\eta^{n+\varphi}$, or the implicit change in pressure, δP , must be known in order to solve equation (3.79). The requirement that either $\rho \vec{u}_\eta^{n+\varphi}$ or δP to be known along the boundary presents little problem. For the different types of boundary conditions described in Section 2.5, either the boundary pressure or the normal velocity at the boundary is specified. The one exception is the subsonic inlet with specified stagnation pressure and temperature with specified direction of flow (see Section 2.5.5.2). In this case, neither the boundary pressure or the normal velocity is specified. However, the relationship between the two is known through the stagnation relations, equations (1.7) and (1.9). The procedure for the PCICE-FEM scheme in applying this boundary condition is to use the explicit predictor value of velocity to determine the boundary pressure with equation (1.9). This pressure is then used to specify δP on the inlet boundary. Best results are obtained when the inlet boundary is positioned sufficiently far enough away from the main flow features.

There is one last point to be made about the treatment of boundary conditions for the PCICE-FEM scheme. It has to do with the determination and use of the outward pointing unit normal vector to the boundary, $\vec{\eta}$. For most second-order partial differential equations, the finite element discretization results in a boundary integral in terms of a normal gradient of the dependent variable. For linear triangular finite elements, a gradient is constant in an element and, thus, is constant along the element face adjacent to the boundary. Also, the outward normal direction is also constant along the boundary face (see Section A.3). On the other hand, the boundary integral contained in equation (3.79) is in terms of the nodal values of the implicit time-weighted mass flux, $\rho \vec{u}_\eta^{n+\varphi}$. Therefore, nodal values of the unit normal vector are required to evaluate the boundary integral in equation (3.79). Nodal outward unit normal vectors differ from the boundary face outward unit normal vectors along curved boundaries. In terms of nodal values, $\rho \vec{u}_\eta^{n+\varphi}$ and $\vec{\eta}$ may vary linearly along a boundary face for linear finite elements.

Determination of the boundary node outward pointing unit normal vector in a consistent manner is crucial to the performance of the PCICE-FEM scheme. For the mass-momentum coupling procedure employed here, the resulting pressure Poisson equation is essentially a mass conservation equation. Therefore, it follows that the principle of global mass conservation should be used to define a unique direction for the outward pointing unit normal vector at any node on the boundary of a discretized domain. For the PCICE-FEM scheme, it is important that the explicit components and the semi-implicit pressure-correction components for the balance of momentum equations adhere to the principle of mass conservation. For example, along a curved free-slip solid wall, the convective mass flux is zero, $\rho \vec{u}_\eta^{n+\varphi} = 0$.

However, the divergence of the explicit balance of momentum components constitute a major portion of the right-hand side of the pressure Poisson equation. The nodal values of $\widetilde{\rho\vec{u}}^{n+\phi}$ are not zero for a free-slip solid wall, just the normal component has been specified to be zero. Therefore, in the convective mass flux volume integral of equation (3.79), the element along a curved free-slip solid wall must be composed of nodal values of $\widetilde{\rho u}^{n+\phi}$ and $\widetilde{\rho v}^{n+\phi}$ that are strictly tangent to the solid wall. Early in this research, the nodal values of $\vec{\eta}$ were determined by face-length averaging. This apparently created a situation along free-slip solid walls where the principle of mass conservation was not adhered to. Non-physical waves were propagated into the domain from the curved free-slip solid wall. Engelman et al. (1982) presented a procedure for the determination of nodal outward pointing unit normal vectors on discretized finite element boundaries that is consistent with the principle of mass conservation. This procedure is summarized in Appendix B. With the implementation of Engelman's procedure for determining $\vec{\eta}$, the non-physical waves are eliminated.

3.3.4 Solving the Pressure Poisson Equation with Bi-CGSTAB

The discretized pressure Poisson equation for the PCICE-FEM scheme, equation (3.79), requires that a linear algebraic system of equations be solved simultaneously. The coefficient matrix derived from the spatial discretization of the governing partial differential equations is sparse. Also, the substitution of the equation of state for the partial derivative of density with respect to time has rendered the coefficient matrix to be non-symmetric. The preconditioned Bi-Conjugate Gradient Stabilized (Bi-CGSTAB) iterative method (Van Der Vorst, 1992) was chosen for solving the discretized pressure Poisson equation for the PCICE-FEM scheme because of its ability to solve a non-symmetric system of equations and for its smooth convergence. A new Jacobi preconditioner was developed for Bi-CGSTAB based upon the coefficient matrix for the pressure Poisson equation. With this new diagonal preconditioner and exact finite element integration, no matrices are required by Bi-CGSTAB, only the assembly of matrix-vector dot products. Also, this new preconditioner has proven to be very efficient for Bi-CGSTAB, requiring only 4-7 iterations to achieve a reduction of four orders of magnitude in the relative error for pressure change. In this section, the Bi-CGSTAB iterative method and the newly developed preconditioner is discussed for the solution of the discretized pressure Poisson equation for the PCICE-FEM scheme.

Equation (3.79) may be recast in the standard form of an algebraic system of equations,

$$Ax = b. \quad (3.81)$$

In equation (3.81), x is the solution vector composed of nodal values of δP . The coefficient matrix A defined by

$$A = \sum_e \int_{\Omega} \{N\}[N] d\Omega \left[\frac{1}{\phi R_c \bar{T}} \right] + \phi \Delta t^2 \sum_e \int_{\Omega} \{\vec{\nabla} N\} \cdot [\vec{\nabla} N] d\Omega \quad (3.82)$$

and b is the forcing vector defined by

$$\begin{aligned}
 b = & \sum_e \int_{\Omega} \{N\}[N] d\Omega \left\{ \frac{\bar{\rho}}{\bar{T}} (\tilde{T} - T^n) \right\} + \Delta t \sum_e \int_{\Omega} \{\vec{\nabla} N\}[N] d\Omega \cdot \{\widetilde{\rho \vec{u}}^{n+\varphi}\} \\
 & - \Delta t \sum_e \int_{\Gamma} \{N\}[N] d\Gamma \{\rho u_{\eta}^{n+\varphi}\} + \Delta t \sum_e \int_{\Omega} \{N\} d\Omega s_e^n \\
 & - \varphi \Delta t^2 \sum_e \int_{\Omega} \{\vec{\nabla} N\} \cdot [\vec{\nabla} N] d\Omega \{P^n\}.
 \end{aligned} \tag{3.83}$$

The coefficient matrix A is a non-symmetric, square matrix dimensioned by the number of nodes in the domain. The A matrix can be extremely sparse. A typical two-dimensional finite element mesh constructed of linear triangular finite elements averages six elements common to a node in the domain. Therefore, a row of A corresponding to a single node has seven locations occupied by finite element contributions from the surrounding elements. This is an average number of locations irrespective of the number of nodes in the domain. The non-symmetric nature of A is due to the equation of state term used to express the time rate of change in density in terms of the implicit change in pressure. This property of A may be explicitly shown by examining a single element's contribution to A . Integrating the first term of equation (3.82) over a single element e for a linear triangular finite element, gives

$$\int_{\Omega} \{N\}[N] d\Omega_e \left[\frac{1}{\varphi R_c \bar{T}} \right] = \frac{\Omega_e}{12\varphi R_c} \begin{bmatrix} \frac{2}{\bar{T}_1} & \frac{1}{\bar{T}_2} & \frac{1}{\bar{T}_3} \\ \frac{1}{\bar{T}_1} & \frac{2}{\bar{T}_2} & \frac{1}{\bar{T}_3} \\ \frac{1}{\bar{T}_1} & \frac{1}{\bar{T}_2} & \frac{2}{\bar{T}_3} \end{bmatrix}, \tag{3.84}$$

where Ω_e is the area of element e . The subscripts on the time-weighted explicit temperature correspond to local node numbers. Equation (3.84) was integrated with the exact integration formula found in equation (0.20) of Appendix A.

A full description of iterative methods for linear systems will not be presented here. There are many methods and the research in this field is currently quite active. At the time of this writing, the preconditioned Bi-CGSTAB algorithm appears to be rapidly gaining popularity as a method for non-symmetric linear systems derived from CFD applications. Duplicating Van Der Vorst's notation, the Bi-CGSTAB algorithm for non-symmetric linear system, equation (3.81), is as follows:

$$\begin{aligned}
 & x_0 \text{ is an initial guess;} \\
 & r_0 = b - Ax_0 \\
 & \rho_0 = \alpha = \omega_0 = 1; \\
 & v_0 = p_0 = 0; \\
 & \text{for } i = 1, 2, 3, \dots;
 \end{aligned}$$

```

 $\rho_i = (r_0, r_{i-1});$ 
 $\beta = (\rho_i / \rho_{i-1})(\alpha / \omega_{i-1});$ 
 $p_i = r_{i-1} + \beta (p_{i-1} - \omega_{i-1} v_{i-1});$ 
Solve  $y$  from  $Ky = p_i$ ;
 $v_i = Ay$ ;
 $\alpha = \rho_i / (r_0, v_i);$ 
 $s = r_{i-1} - \alpha v_i$ ;
Solve  $z$  from  $Kz = s$ ;
 $t = Az$ ;
 $\omega_i = (t, s) / (t, t);$ 
 $x_i = x_{i-1} + \alpha y + \omega_i z$ ;
if  $x_i$  is accurate enough, then stop;
 $r_i = s - \omega_i t$ ;
end

```

For the PCICE-FEM scheme, the initial estimate for the solution vector comes from the explicit predictor for pressure,

$$x_0 = \varphi(\tilde{P} - P^n).$$

In the Bi-CGSTAB algorithm, Van Der Vorst (1992) never describes the procedure of applying Dirichlet boundary conditions in Bi-CGSTAB. In a personal communication with Van Der Vorst (1999), he alluded that the system of equations is modified to include Dirichlet boundary conditions. This procedure would require storage of A . Storage of a large square matrix is undesirable for a FEM scheme where it is far more efficient to assemble and store the dot product of A with any vector. The procedure developed here is to apply the Dirichlet boundary conditions on x_0 and set y , and z equal to zero. This forces the iterated x_i to equal x_0 . Convergence for the Bi-CGSTAB algorithm is achieved when the L_2 -norm of the ratio of the iterated residual to the initial residual is

$$\left\| \left\{ \frac{r_i}{r_0} \right\} \right\|_{L_2} \leq 1.0 \times 10^{-4}. \quad (3.85)$$

The matrix K in the above algorithm is the preconditioning matrix. While the Bi-CGSTAB algorithm may be employed without a preconditioner, there are two main reasons for preconditioning. The first is that preconditioning reduces the computational effort required to solve the linearized system of equations. And second, preconditioning reduces the total number of iterations for convergence. Without preconditioning, Bi-CGSTAB is only guaranteed to converge in the number of iterations equal to the dimension of A . There are many preconditioners for linear systems and will not be discussed here. The preconditioning matrix developed for the PCICE-FEM scheme is a Jacobi (diagonal) matrix. Jacobi preconditioning matrices are not known for rapid convergence. However, they are easily

stored in a vector matrix and are inexpensive to implement in the above algorithm as the inversion of a diagonal matrix is trivial. The desire for the PCICE-FEM scheme was to develop a Jacobi preconditioning matrix that would converge to the criteria of equation (3.85) in 10 iterations or less. A Jacobi preconditioning matrix was found that fits this criteria. Defining two sub-matrices of A as

$$M^\rho = \sum_e \int_\Omega \{N\}[N] d\Omega \left[\frac{1}{\phi R_c \bar{T}} \right] \quad (3.86)$$

and

$$K^\rho = \phi \Delta t^2 \sum_e \int_\Omega \{\vec{\nabla} N\} \cdot [\vec{\nabla} N] d\Omega, \quad (3.87)$$

the new Jacobi preconditioning matrix for the PCICE-FEM scheme is defined as the lumped-mass form of M^ρ plus the diagonal of K^ρ ,

$$K = M_l^\rho + \text{diag } K^\rho. \quad (3.88)$$

This new preconditioning matrix allows convergence of the Bi-CGSTAB algorithm in approximately 4 - 7 iterations for the PCICE-FEM scheme.

3.4 Pressure-Correction of the Semi-Implicit Conservation Equations

The final phase of the PCICE algorithm is the pressure-correction of the explicit predictor phase flow variables. This is accomplished by solving a derived set of pressure-correction equations that represent the difference between the implicit and the explicit flow variables plus a pressure-correction. This difference is derived by subtracting the explicit predictor phase equations from the semi-implicit equations. At this point, the explicit equations are expressed in terms of the time-advanced explicit values smoothed by artificial dissipation. The pressure-correction is based upon the implicit pressure field obtained with the solution of the pressure Poisson equation. This section is devoted to the final algorithmic phase of the PCICE-FEM scheme. The derivation of the pressure-correction equations will be presented first. The finite element discretization and solution of the pressure-correction equations will then follow.

3.4.1 Derivation of the Pressure-Correction Equations

The derivation of the pressure-correction equations for the PCICE algorithm is a straightforward process of subtracting the explicit predictor phase equations that are expressed in terms of the time-advanced explicit values smoothed by artificial dissipation from the semi-implicit discretization of the governing hydrodynamic equations. The

derivation begins by considering the semi-implicit balance of momentum equations, equation (3.12). Note that this equation is already in pressure-correction form. The only implicit variable on the right-hand side of equation (3.12) is a known quantity resulting from the solution of the pressure Poisson equation. During the derivation of the pressure Poisson equation (see Section 3.3.1), equation (3.12) has been previously expressed in terms of the smoothed explicit time-advanced momentum components by equation (3.50). By recasting equation (3.50) in terms of δP , the final form of the pressure-correction equation for the implicit balance of momentum components is

$$\rho \vec{u}^{n+1} = \widetilde{\rho \vec{u}} - \Delta t \vec{\nabla}(\delta P + P^n). \quad (3.89)$$

The correction equation for mass conservation is derived by subtracting equation (3.14) from equation (3.6), yielding

$$\rho^{n+1} = \rho^n + \delta \tilde{\rho} - \Delta t \vec{\nabla} \cdot (\phi \rho \vec{u}^{n+1} - \phi \delta \rho \vec{u} - \phi \rho \vec{u}^n). \quad (3.90)$$

The explicit change in mass, defined in equation (3.15), is also expressed in terms of values smoothed by artificial dissipation,

$$\delta \tilde{\rho} = \tilde{\rho} - \rho^n. \quad (3.91)$$

By substituting equations (3.51) and (3.91) into equation (3.90), the difference between the mass conservation semi-implicit and explicit predictor equations becomes the final correction equation for mass conservation,

$$\rho^{n+1} = \tilde{\rho} - \phi \Delta t \vec{\nabla} \cdot (\rho \vec{u}^{n+1} - \widetilde{\rho \vec{u}}). \quad (3.92)$$

While this equation is not strictly speaking a pressure-correction equation, the correction term on the right-hand side is a result of correcting the balance of momentum components for the implicit pressure. This correction equation for mass conservation is a different approach for correcting mass from that used in the original ICE scheme of Harlow and Amsden (1971) where the equation of state was employed to correct mass. The correction term on the right-hand side of equation (3.92) is essentially the right-hand side of the pressure Poisson equation given by equation (3.54). The correction for mass given by the equation of state, equation (3.55), is essentially the left-hand side of equation (3.54). Both of these correction approaches for mass should be mathematically equivalent. However, from a numerical standpoint, they are applied in very different manners. A correction equation for mass derived from an equation of state is an algebraic equation applied at a point. The correction equation for mass given by equation (3.92) requires spatial discretization. Both of these correction approaches for mass were evaluated for the PCICE-FEM scheme with the correction approach of equation (3.92) giving far superior results. Equation (3.92) has the advantage of satisfying mass

conservation requirements in a global manner. Also, there is no time-advanced energy requirement in equation (3.92) as there is in equation (3.55).

The correction equation for total energy is derived by subtracting equation (3.16) from equation (3.8) to give

$$\rho e_t^{n+1} = \rho e_t^n + \delta \rho e_t - \Delta t \vec{\nabla} \cdot \left[\phi \rho \vec{u}^{n+1} H^{n+1} + (1 - \phi) \rho \vec{u}^n H^n - (\phi \delta \rho \vec{u} + \rho \vec{u}^n) H^n \right], \quad (3.93)$$

where H^{n+1} is defined by equation (3.9). The explicit change in total energy, defined in equation (3.17), is also expressed in terms of values smoothed by artificial dissipation,

$$\delta \rho e_t = \widetilde{\rho} e_t - \rho e_t^n. \quad (3.94)$$

By substituting equations (3.51) and (3.94) into equation (3.93), the difference between the total energy semi-implicit and explicit predictor equations becomes the final correction equation for total energy,

$$\rho e_t^{n+1} = \widetilde{\rho} e_t - \phi \Delta t \vec{\nabla} \cdot (\rho \vec{u}^{n+1} H^{n+1} - \widetilde{\rho} \vec{u} H^n). \quad (3.95)$$

3.4.2 Finite Element Discretization and Solution of the Pressure-Correction Equations

In the PCICE-FEM scheme, the pressure-correction equations are spatially discretized with the standard Galerkin FEM. They are solved in successive order of equations (3.89), (3.92), and (3.95) so that all of the right-hand side terms are known nodal quantities. Thus, integration by parts is not necessary because the boundary conditions at this point are already satisfied and there is no problem with differentiability constraints for nodal values. To begin the development of the finite element discretization of the pressure-correction equations, equations (3.89), (3.92), and (3.95) may be recast in correction form as

$$U' = -\Delta t \vec{\nabla} \cdot \vec{F}', \quad (3.96)$$

where

$$U' = \left\{ \begin{array}{c} \rho^{n+1} - \widetilde{\rho} \\ \rho \vec{u}^{n+1} - \widetilde{\rho} \vec{u} \\ \rho e_t^{n+1} - \widetilde{\rho} e_t \end{array} \right\} \quad (3.97)$$

and

$$\vec{F}' = \begin{Bmatrix} \varphi(\rho \vec{u}^{n+1} - \widetilde{\rho \vec{u}}) \\ \delta P - P^n \\ \varphi(\rho \vec{u}^{n+1} H^{n+1} - \widetilde{\rho \vec{u}} H^n) \end{Bmatrix}. \quad (3.98)$$

U' and \vec{F}' are defined as the correction variables and correction fluxes, respectively. The weighted residual form of equation (3.96) is

$$\int_{\Omega} U' W d\Omega = -\Delta t \int_{\Omega} \vec{\nabla} \cdot \vec{F}' W d\Omega, \quad (3.99)$$

where W is some weight function with appropriate interpolation properties. Denoting the piecewise linear shape function associated with node j by N_j , the finite element approximations for equation (3.99) are

$$U' \approx \sum_j U'_j N_j = [U']\{N\} \text{ and } \vec{F}' \approx \sum_j \vec{F}'_j N_j = [\vec{F}']\{N\}. \quad (3.100)$$

where the sum over j is the sum over all of the nodes in the domain.

Applying the standard Galerkin approximation, $W = N_j$, and substituting the above finite element approximations into equation (3.99), yields the final form of the finite element discretized correction equation,

$$\sum_e \int_{\Omega} \{N\} [N] d\Omega \{U'\} = -\Delta t \sum_e \int_{\Omega} \{N\} [\vec{\nabla} N] d\Omega \cdot \{\vec{F}'\}, \quad (3.101)$$

where the sum over e is the sum over all of the elements in the domain. Substituting equations (3.97) and (3.98) into equation (3.101) yields the discretized correction equation for each hydrodynamic variable,

$$\sum_e \int_{\Omega} \{N\} [N] d\Omega \{\rho^{n+1} - \widetilde{\rho}\} = -\varphi \Delta t \sum_e \int_{\Omega} \{N\} [\vec{\nabla} N] d\Omega \cdot \{\rho \vec{u}^{n+1} - \widetilde{\rho \vec{u}}\}, \quad (3.102)$$

$$\sum_e \int_{\Omega} \{N\} [N] d\Omega \{\rho \vec{u}^{n+1} - \widetilde{\rho \vec{u}}\} = -\Delta t \sum_e \int_{\Omega} \{N\} [\vec{\nabla} N] d\Omega \{\delta P - P^n\}, \quad (3.103)$$

and

$$\sum_e \int_{\Omega} \{N\} [N] d\Omega \{\rho e_t^{n+1} - \widetilde{\rho e_t}\} = -\varphi \Delta t \sum_e \int_{\Omega} \{N\} [\vec{\nabla} N] d\Omega \cdot \{\rho \vec{u}^{n+1} H^{n+1} - \widetilde{\rho \vec{u}} H^n\}. \quad (3.104)$$

To solve the discretized correction equations, equations (3.102) to (3.104) are cast in the form of equation (2.13) and iteratively solved with equation (2.17).

3.4.3 The PCICE-FEM Algorithmic Steps

The algorithmic steps for the PCICE-FEM scheme are as follows:

1) Beginning of time step.

Explicit Predictor Phase

- 2) Solve equation (3.18) for elemental values of $\rho^{n+\varphi}$ and $\rho\vec{u}^{n+\varphi}$.
- 3) Solve equation (3.21) for $\rho\vec{u}^*$.
- 4) Solve equation (3.33) for ρ^* and ρe_t^* .
- 5) Apply artificial dissipation to the explicit components to give $\tilde{\rho}$, $\tilde{\rho\vec{u}}$, and $\tilde{\rho e_t}$.
- 6) Determine the explicit pressure and temperature fields, \tilde{P} and \tilde{T} , using the equation of state.

Elliptic Pressure Poisson Solution

- 7) Determine pressure Poisson equation forcing function b from equation (3.83).
- 8) Solve the pressure Poisson equation, equation (3.79), for δP with Bi-CGSTAB.
- 9) Compute the implicit pressure field P^{n+1} from δP using equation (3.53),

$$P^{n+1} = \frac{\delta P}{\varphi} + P^n.$$

Semi-Implicit Pressure Correction

- 10) Solve equation (3.103) to obtain $\rho\vec{u}^{n+1}$.
- 11) Solve equation (3.102) to obtain ρ^{n+1} .
- 12) Solve equation (3.104) to obtain and ρe_t^{n+1}
- 13) Determine T^{n+1} with equation of state.

14) Next time step.

3.5 Stability of the PCICE-FEM Scheme

An in-depth stability analysis was not performed for the PCICE-FEM scheme. Typically, these stability analyses are performed upon linear systems of equations and are approximations at best. It was expected that, because the two-step Taylor-Galerkin FEM scheme is the basis for the explicit predictor for the balance of momentum, the coefficient α in the stability criteria given by equation (2.1) would remain the same. With the semi-implicit treatment of the governing hydrodynamic equations for the PCICE-FEM scheme, the acoustic component of (2.1) is eliminated. Thus, the time step size for the PCICE-FEM scheme is governed by the stability criteria of

$$\Delta t \leq \frac{\alpha \beta h_e}{|\vec{u}|}, \quad (3.105)$$

for $\varphi \geq 0.5$. This stability criteria has proven to work quite well for the PCICE-FEM scheme when simulating inviscid flows. A stability coefficient of $\alpha = 0.5$ and a safety factor of $\beta = 0.8-0.9$ are the parameters used for transient inviscid simulations. The safety factor may be increased up to $\beta = 1.2$ for steady-state simulations when residual smoothing is performed.

For viscous simulations, the same approach as above was taken to determine the stability criteria for the PCICE-FEM scheme. Morgan (1987) based the viscous stability criteria for the two-step Taylor-Galerkin FEM scheme upon the cell Peclet number. Removing the acoustic component, Morgan's viscous stability criteria gives

$$\Delta t \leq \frac{\alpha \beta h_e^2 c_p \rho}{2k + |\vec{u}| h_e c_p \rho}, \quad (3.106)$$

for $\varphi \geq 0.5$. This criteria works well with the PCICE-FEM scheme for low to high-speed viscous flows. For very slow viscous flows (see Section 4.2.3), β must be reduced considerably, on the order of $\beta = 0.1$.

4. Results and Conclusions

A new computational fluid dynamics (CFD) scheme may be formulated with the most accurate numerical approximations and with the fewest simplifying assumptions incorporated into the governing hydrodynamic equations, but the real test of the new scheme is in the quality of the results that it produces. This is sometimes difficult to quantify as individual CFD users tend to have different requirements. Typically, extreme solution accuracy comes at an excessive cost in computational resources. In today's world, the overall run times can be critical to a project's success, especially for sensitivity analyses where many test cases may need to be analyzed. The ideal compromise for a CFD simulation is the highest possible solution accuracy for a moderate computational cost. This is the main driving force behind the development of the PCICE-FEM scheme. Simply stated, the sole purpose of developing the PCICE-FEM scheme is to provide the most accurate and efficient simulation results to date for the widest range of compressible flows.

The first section of this chapter is devoted to documenting the solution capability of the PCICE-FEM scheme. Six simulations with flow regimes ranging from nearly incompressible ($\text{Mach} = 0.0006$) to highly compressible ($\text{Mach} = 2.7$) viscous and inviscid flows are illustrated. The first three simulations demonstrated here are of transient flows designed to demonstrate the PCICE-FEM scheme's ability to accurately propagate wave forms. The first of which is the classic one-dimensional shock tube problem for which there is an (inviscid) analytical solution. The second inviscid transient problem is an idealized simulation of a two-dimensional accidental laboratory steam explosion. The final transient simulation is low-speed viscous (laminar) two-dimensional flow around a cylinder resulting in an oscillatory flow pattern downstream of the cylinder. Next, three steady-state simulations are presented which are designed to demonstrate the PCICE-FEM scheme's ability to accurately converge to a unique solution. The first steady-state simulation is of an inviscid supersonic flow in a channel with an 18 degree wedge-shaped obstruction that results a complex shock wave reflection pattern. The second steady-state simulation is a viscous (laminar) double-throated converging-diverging nozzle with flows varying from $\text{Mach} = 0.01$ to $\text{Mach} = 2.7$. The final steady-state simulation is of thermally driven flow in a square cavity for three different Rayleigh numbers.

A FORTRAN computer code employing the PCICE-FEM scheme has been constructed for validation purposes and for computing the results provided in this chapter. The code is spatially two-dimensional, incorporating the governing hydrodynamic equations found in equation (1.1). Domain discretization and spatial integration is accomplished by linear triangular finite elements. The non-dimensional form of the governing hydrodynamic equations was not employed, nor was there any effort made to normalize or scale the dependent hydrodynamic variables. Thus, variables with a consistent set of units is required in the computations. All of the simulations in this chapter were run on a 1.0 GHz AMD K7 personal computer with 256 Mb of memory.

The system of units used for the simulations presented in this chapter is the International System, more commonly referred to as SI units. The basic SI units of mass, length, and time for these simulations, are respectively, the kilogram (kg), meter (m), and seconds (s). The unit

of force is the Newton ($1\text{N} = 1\text{ kg}\cdot\text{m/s}^2$) and the unit for pressure is called the Pascal ($1\text{ Pa} = 1\text{ N/m}^2 = 1\text{ kg}\cdot\text{m/s}^2$). The standard atmosphere is used frequently in the following simulations and is defined as $1\text{ atm.} = 101325.0\text{ Pa}$. The scale for measuring temperature in SI units is the Celsius scale ($^{\circ}\text{C}$) and is used in conjunction with the absolute scale, the Kelvin (K).

With out normalization of the governing hydrodynamic equations, the viscous simulations with a specified Reynolds or Rayleigh number (non-dimensional parameters describing a state of flow) require that either the characteristic length or the dynamic viscosity be scaled. The dimensionless Reynolds number is stated as the ratio of inertia force to viscous force and is defined (e.g. White, 1991) as

$$Re = \frac{\rho UL}{\mu}, \quad (4.1)$$

where U is a velocity scale, L is a characteristic geometric size, and ρ and μ are the fluid density and dynamic viscosity. The dimensionless Rayleigh number is stated as the ratio of buoyancy force to change of momentum flux and is defined (e.g. Arpaci, 1984) as

$$Ra = \frac{g \rho^2 \beta_T \Delta T L^3}{\mu^2} Pr, \quad (4.2)$$

where g is the gravitational constant, ΔT is the change in temperature of the region in question (such as in a hot vertical plate and the free-stream fluid temperature), and Pr is another dimensionless parameter called the Prandtl number ($Pr = 0.72$ for air). β_T is the coefficient of thermal expansion defined (e.g. Arpaci, 1984) by

$$\beta_T = -\frac{1}{\rho} \frac{\partial \rho}{\partial T} \bigg|_P, \quad (4.3)$$

or for an ideal gas,

$$\beta_T = -\frac{1}{\rho} \frac{P}{T^2 R_c}. \quad (4.4)$$

For problems requiring specified Re or Ra , the length was chosen arbitrarily for convenience and the dynamic viscosity was varied accordingly. For the laminar viscous simulations presented here, the dynamic viscosity and coefficient of thermal conductivity k are defined as domain constants. The coefficient of thermal conductivity can be approximately related to the dynamic viscosity (e.g. Hoffman et al., 1996) by

$$k = c_P \frac{\mu}{Pr}. \quad (4.5)$$

Five of the simulations employ the ideal gas assumption, equation (1.3), for air with $\gamma = 1.4$ and $R_c = 287.04 \text{ (N}\cdot\text{m)/(kg}\cdot\text{K)}$. The second transient simulation is of the laboratory accident scenario involving a rapid vaporization of a small volume of water. For this problem illustrated in Section 4.1.2, the ideal gas behavior was assumed for super-heated steam.

This final chapter will conclude with a section describing the research conducted here. This will begin with a description of the PCICE-FEM scheme. From this description, conclusions will be drawn pertaining to the novel contributions made to the field of CFD, specifically in the area of semi-implicit pressure-based schemes. The ability of the PCICE-FEM scheme to solve a wide range of flow regimes will then be discussed. Statements will be made about the effectiveness of the PCICE-FEM scheme for the different flow regimes. Finally, recommendations for further improvements to the method will be stated.

4.1 Transient Simulations

The first of three transient simulations to be illustrated here is the shock tube which is a Riemann type problem. It is an inviscid one-dimensional problem that is commonly used to validate new schemes and codes. It is one of the few compressible flow problems that has an analytical solution. The second transient simulation is that of a laboratory accident involving a steam explosion resulting from a very rapid vaporization of water dropped into a hot crystallizing dish causing the destruction of the crystallizing dish, a variable temperature hot plate, and a beaker containing a liquid chemical mixture. The final transient simulation presented here is of low-speed viscous flow around a cylinder. The laminar flow around the cylinder results in a downstream oscillatory flow field known as the Von Karmen vortex street. The transient simulations in this section are run second-order temporally accurate with the time-weighting parameter for the PCICE-FEM formulation set to $\phi = 0.5$.

4.1.1 The Shock Tube Problem

The shock tube problem is a classic test case for the validation of compressible flow schemes and codes. It is one of the few compressible flow problems that has an analytical solution available (e.g. Hirsch, 1990) that can be compared to numerical and experimental results. The problem can be characterized by the sudden rupture of a diaphragm in a long one-dimensional tube separating two initial gas states at different pressures and densities (see the initial conditions in Figure 4-1). The state after the rupture of the diaphragm ($t > 0$) consists of three major compressible flow phenomena; a shock wave propagating downstream at a speed equal to $|\vec{u}| + c$, followed by a contact discontinuity traveling with the fluid at a velocity of $|\vec{u}|$, and an expansion fan (rarefaction wave) traveling upstream at a rate of $|\vec{u}| - c$. The interesting aspect of the shock tube problem is that all of the above major flow phenomena associated with compressible flow are concurrently present. The results of the characteristic analysis performed in Section 2.5 become readily apparent as these flow phenomena are easily identified propagating along the characteristic paths defined by the eigenvalue solution of equation (2.58).

Figure 4-1 depicts the initial conditions of the shock tube problem at time $t'' = 0.0$ with a 10 to 1 isothermal pressure drop across the diaphragm located at $x = 40.0$ m. The two-dimensional finite element mesh for this one-dimensional problem is constructed of a 101×3 node structured grid arrangement with a 1.0 m spacing. The resulting mesh then consists of 303 nodes and 400 triangular finite elements.

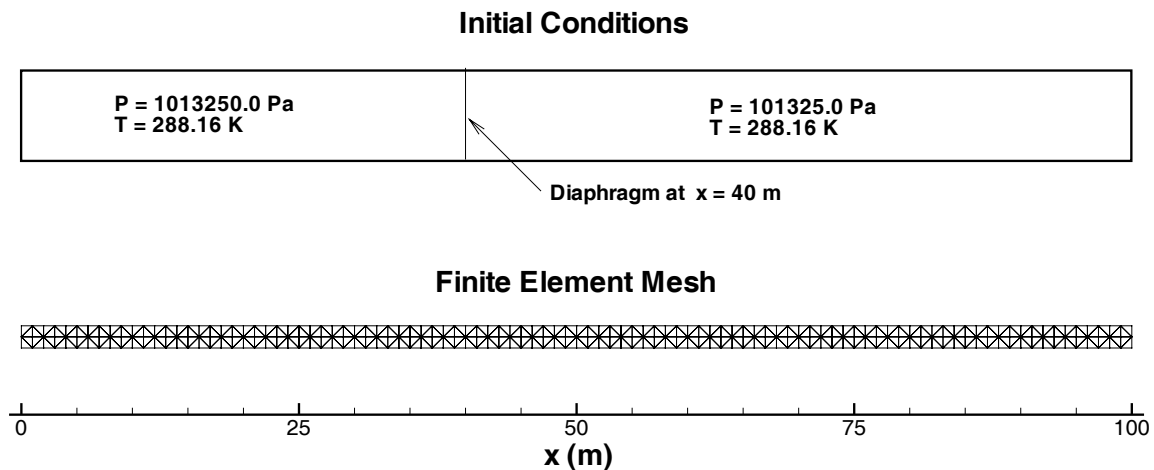


Figure 4-1 Initial conditions and finite element mesh for the shock tube problem.

Figure 4-2 through Figure 4-5 illustrate the Flux-Corrected Transport (FCT) process for the shock tube problem. These plots of density are for the high-order scheme, the low-order scheme, the PCICE-FEM scheme with FCT applied, and the original FEM-FCT scheme all at a solution time of $t = 0.08$ s. Lapidus artificial diffusion was employed for both the PCICE-FEM scheme and the original FEM-FCT scheme to control the nonphysical stair stepping phenomena generated by the FCT in the rarefaction wave.

The plots in Figure 4-2 through Figure 4-5 were created with the nodal values of the variables taken along the center line of the finite element mesh. Figure 4-2 is the high-order PCICE-FEM solution for density obtained with the explicit predictor phase equations solved by equation (2.17) without any form artificial dissipation applied. The high-order spatial effects are clearly manifested in the non-physical numerical oscillations. Figure 4-3 is the low-order PCICE-FEM density solution obtained with the explicit predictor phase equations solved by equation (2.39) employing the diffusion coefficient defined in (3.48). It is interesting to note that neither the high-order or low-order solutions appear to be very accurate. Yet combined properly with the FCT scheme, they yield the very accurate PCICE-FEM density solution shown in Figure 4-4. This semi-implicit FCT solution compares very favorably with the solution shown in Figure 4-5 obtained with the original explicit FEM-FCT scheme. Both FCT solutions yield extremely accurate results for the shock tube problem with the PCICE-FEM solution requiring approximately half the number of time steps as the original FEM-FCT

scheme. The results obtained with the PCICE-FEM scheme appear to resolve the contact discontinuity, located at $x \approx 62$ m, more accurately than those obtained with the FEM-FCT scheme. This is possibly due to the stronger pressure-density coupling of the PCICE-FEM formulation versus a purely explicit scheme, such as the FEM-FCT scheme, where the discretized governing hydrodynamic equations are de-coupled in time.

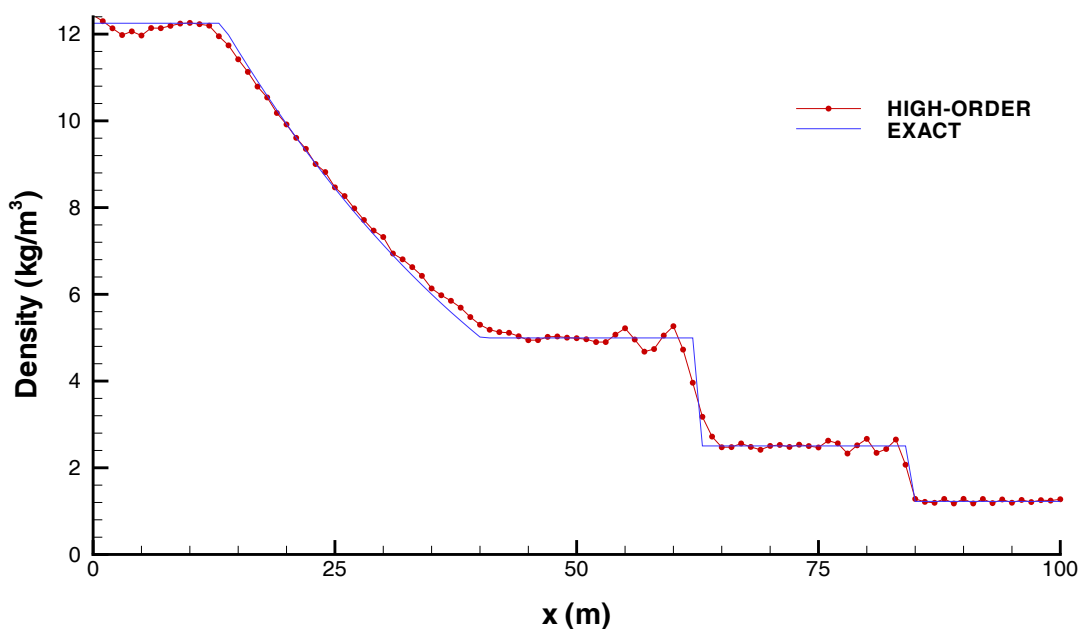


Figure 4-2 High-order PCICE-FEM shock tube solution for density at time $t = 0.08$ s.

Figure 4-6 and Figure 4-7 are the PCICE-FEM shock tube solutions for pressure and velocity, respectively. Across a contact discontinuity, the pressure and velocity are constant while the density and temperature (shown in Figure 4-8) are discontinuous analytically. Again, the numerical solution accuracy when compared to the analytical solution is excellent.

4.1.2 Laboratory Accident Simulation

This simulation was performed as part of a laboratory accident investigation. The motivation for this simulation was the accident investigators desire to document a possible sequence of events leading to minor abdominal injuries sustained by a laboratory technician while collecting experimental results. This simulation is also an opportunity to demonstrate the ability of the PCICE-FEM scheme to propagate multi-dimensional shock waves with FCT.

According to the accident description, the technician began performing a chemical reaction process to synthesize custom silicon chemicals. The required chemicals were placed in a 100 ml round bottom schenk flask and the total volume of chemicals in the flask was approximately 45 ml. After addition of the of the reaction contents, the pressure in the flask

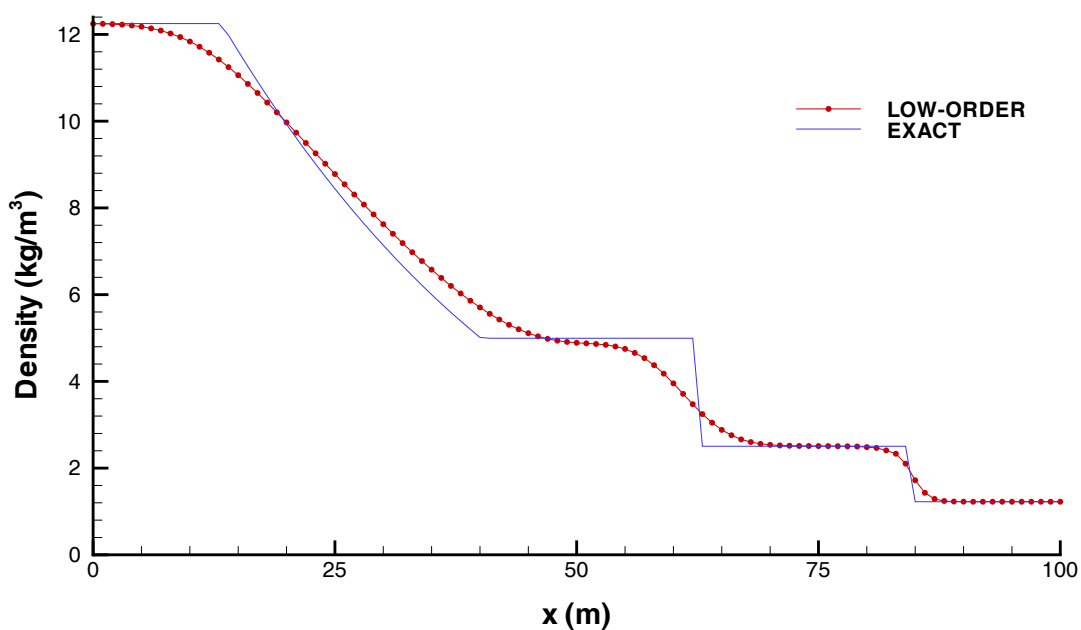


Figure 4-3 Low-order PCICE-FEM shock tube solution for density at time $t = 0.08$ s.

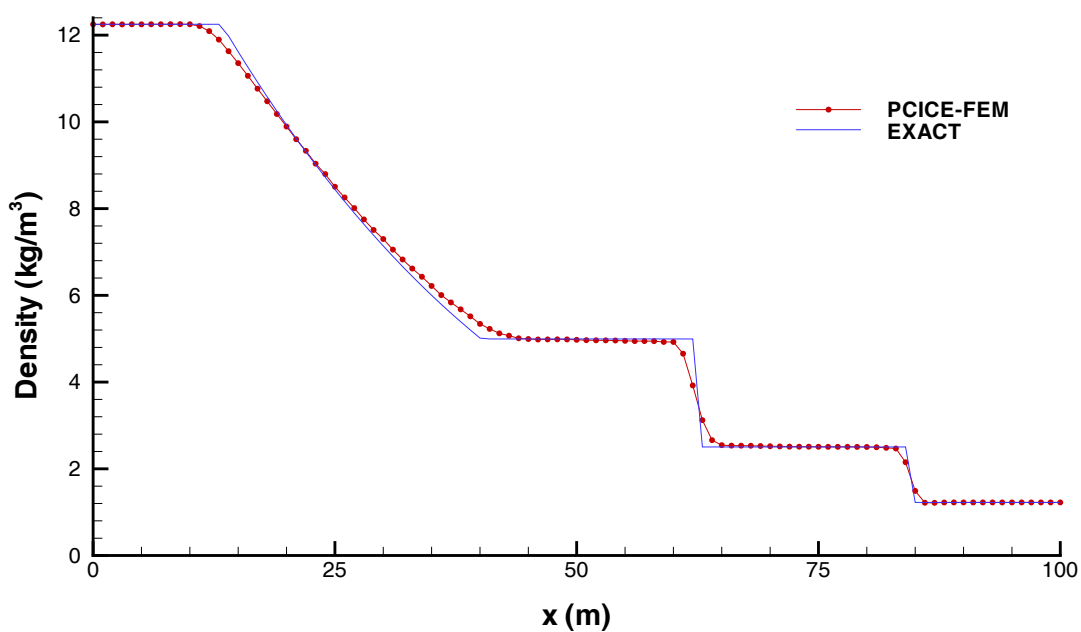


Figure 4-4 PCICE-FEM shock tube solution for density at time $t = 0.08$ s

was equalized with the atmosphere, and the flask was sealed. The 100 ml flask was placed in a 10 cm crystallizing dish, which served as a water bath (deionized water was used) and the bath

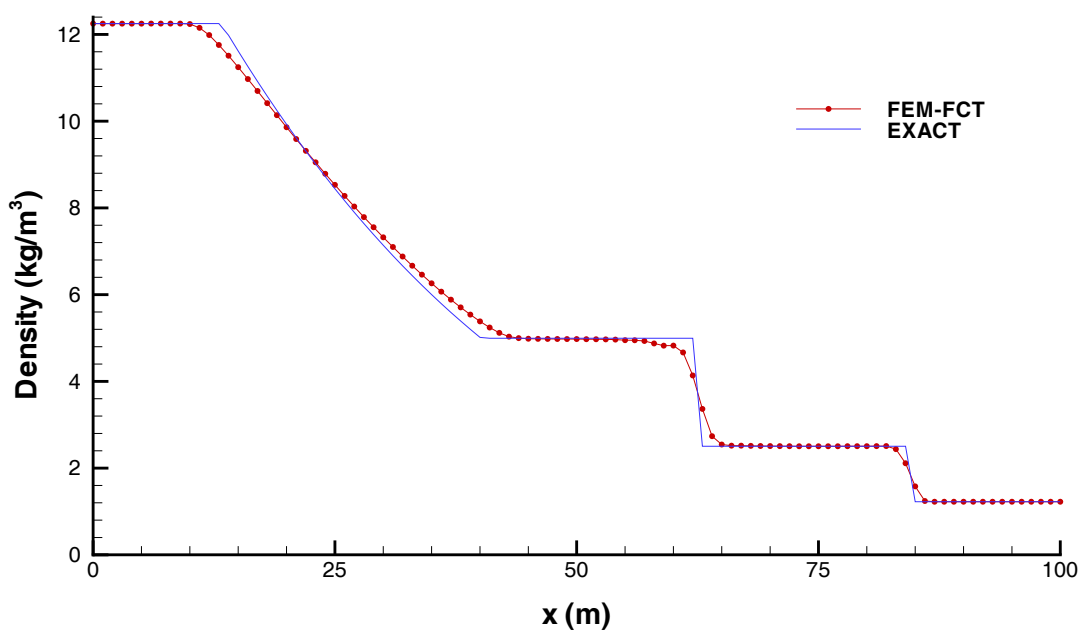


Figure 4-5 FEM-FCT shock tube solution for density at $t = 0.08$ s.

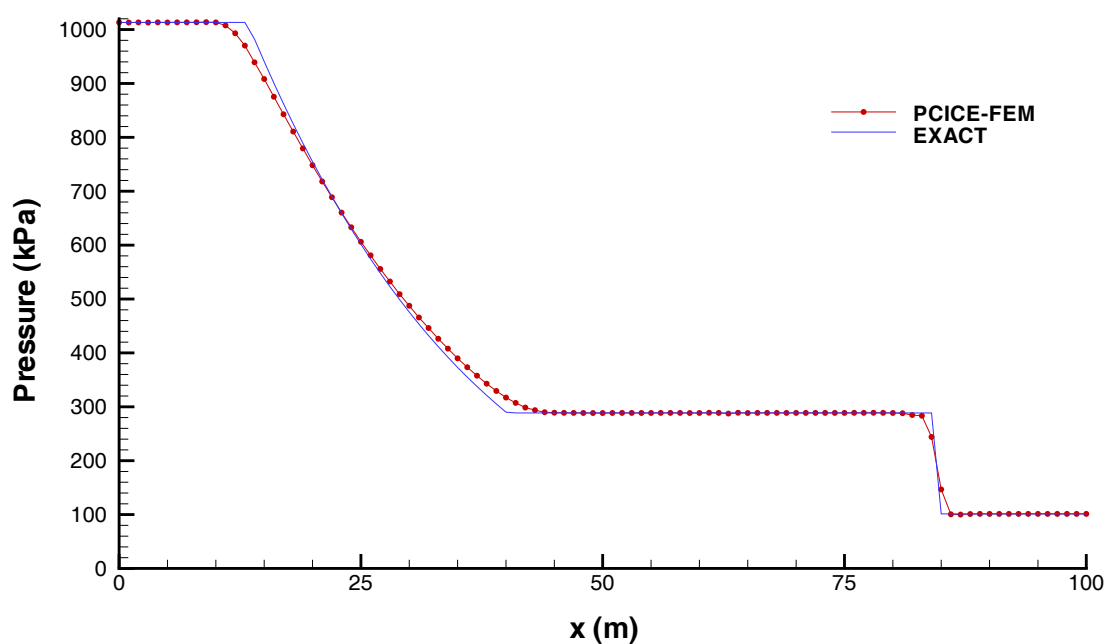


Figure 4-6 PCICE-FEM shock tube solution for pressure at time $t = 0.08$ s.

was magnetically stirred. This assembly was placed on a Corning ceramic top stirring hot plate mounted on a lab jack in a class A fume hood. The heater was turned on and the bath

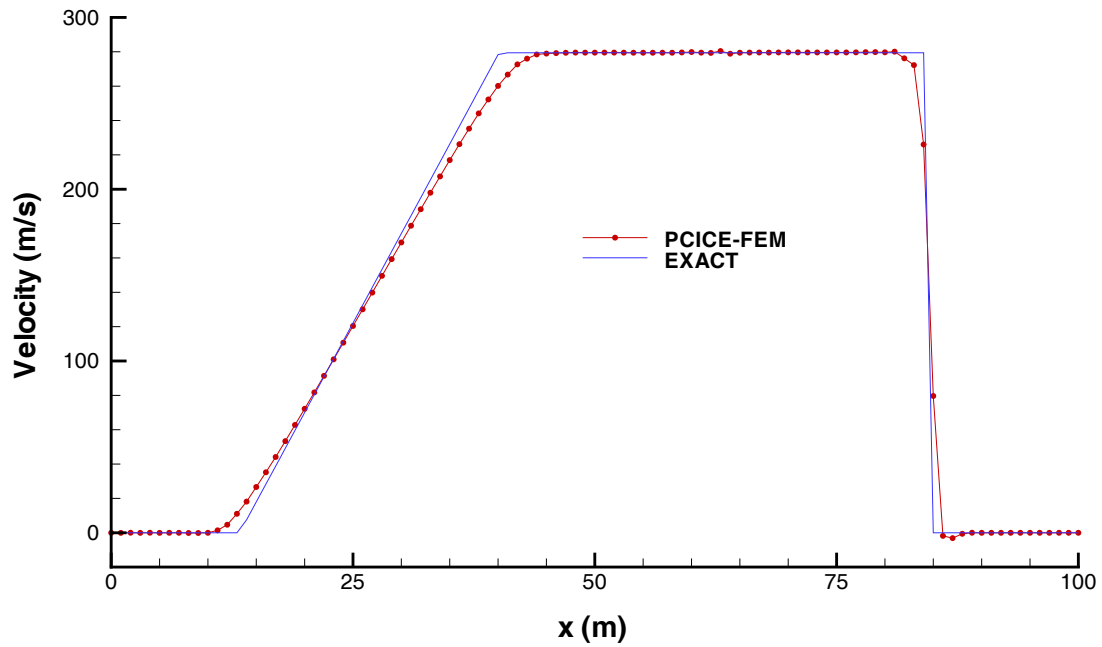


Figure 4-7 PCICE-FEM shock tube solution for velocity at time $t = 0.08$ s.

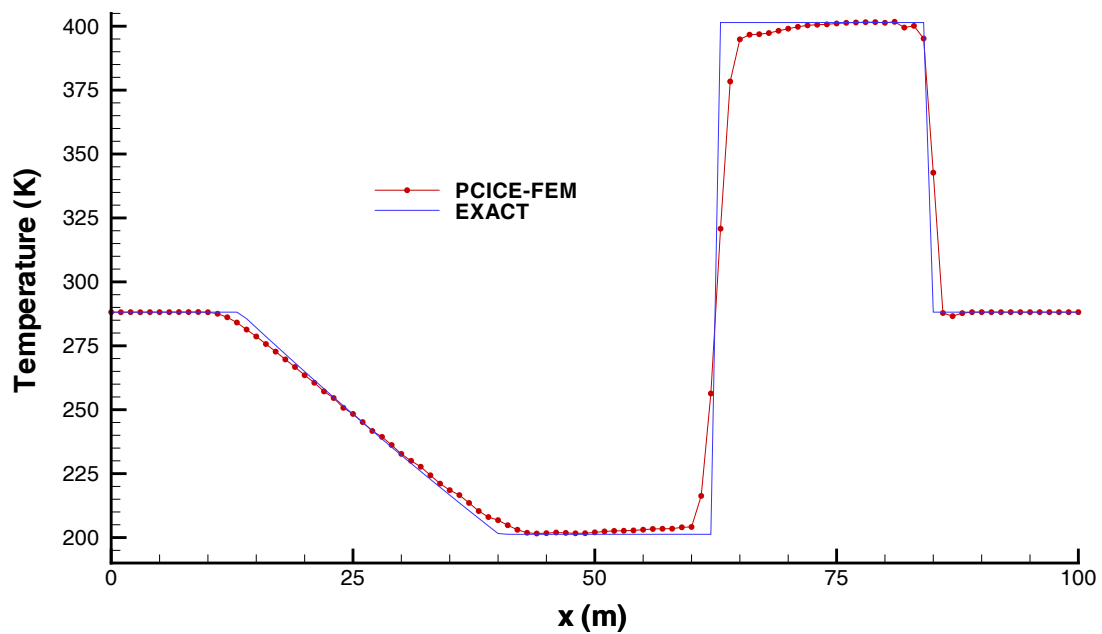


Figure 4-8 PCICE-FEM shock tube solution for temperature at time $t = 0.08$ s.

temperature was monitored with a mercury-filled thermometer. Heating occurred over the next two hours with the water bath stabilized at 58 °C for over an hour. The technician then

left the lab for a one hour lunch. When the technician returned to the lab, he noted that the water in the crystallizing dish had completely evaporated. The technician then decided to refill the dish and attempted to pour deionized water into the dish. At this point, the accident occurred. The ceramic hot plate, the 100 ml flask, the thermometer, and the crystallizing dish were all destroyed. The technician felt no pain but his ears were ringing and there was blood on his shirt.

The most likely scenario leading to the destruction of the laboratory equipment is attributed to the initiation of rapid water vaporization caused by the laboratory technician pouring deionized water onto the crystallizing dish. It is estimated that the hot plate temperature could have been as high as 400 °C (752 °F). Thus, the bottom of the crystallizing dish could have been nearly as high of a temperature as the hot plate. At this assumed temperature, rapid vaporization, or explosion, of the deionized water could have taken place. A sudden vaporization of this type would have resulted in a high-pressure wave impinging upon the 100-ml flask and thereby causing the flask to shatter. It has been estimated that the flask was at a temperature of 300 °C, which is far above the boiling point at 1 atmosphere for all of the chemical components in the flask. This super-heated mixture was computed to have a pressure in the 10-12 atmosphere range (approximately 1000 kPa). The sudden release of this superheated mixture at 1 atmosphere upon the bottom of the crystallizing dish would have resulted in an even larger vapor explosion. This scenario would be consistent with the damage the laboratory equipment suffered.

The simulation presented here is based upon the above assumptions. The domain in question would be better addressed by three-dimensional multi-phase modeling. However, due to time constraints, a two-dimensional single-phase simulation was rapidly performed. A two-dimensional simulation will somewhat over predict the peak pressures the 100 ml flask and crystallizing dish were subjected to but will provide a reasonable upper bound on the peak pressures. The triangular finite element mesh discretization the domain in question is shown in Figure 4-9. The 100 ml flask and crystallizing dish were treated as solid wall boundaries. The domain surface at $x = -0.02$ m is a non-reflecting boundary. The surface at $y = 0.05$ m is open to atmospheric pressure. The region centered around $x = 0.04$ m and $y = 0.0$ m was assumed to be a likely point at which the deionized water vaporized. A volume of water, 8 mm³, was assumed to be instantly vaporized in this region at a pressure of 60000.0 kPa. The rest of the domain was initially set to a uniform pressure of 101325.0 Pa. This constitutes a 600 to 1 pressure drop for initial conditions. The entire domain was initially set to a uniform temperature of 350.0 °C. The fluid was assumed to follow the behavior of ideal gas with the ratio of specific heats and gas constant for super-heated steam, $\gamma = 1.33$ and $R_c = 461.15$ (N·m)/(kg·K), respectively.

The color figures, Figure 4-10 through Figure 4-12,, are snapshots of the transient simulation for pressure. Peak pressure on the flask occurred approximately 25 μ s after vaporization. At this point in time, the computed pressure was 1504.1 kPa. Coupling this pressure with the internal load in the flask and fracturing of the flask would soon follow. It is interesting how low the pressure goes in the rarefaction wave region. At 55 μ s after vaporization (Figure 4-12), the minimum pressure in the rarefaction wave region is 23.5 kPa. This pressure is

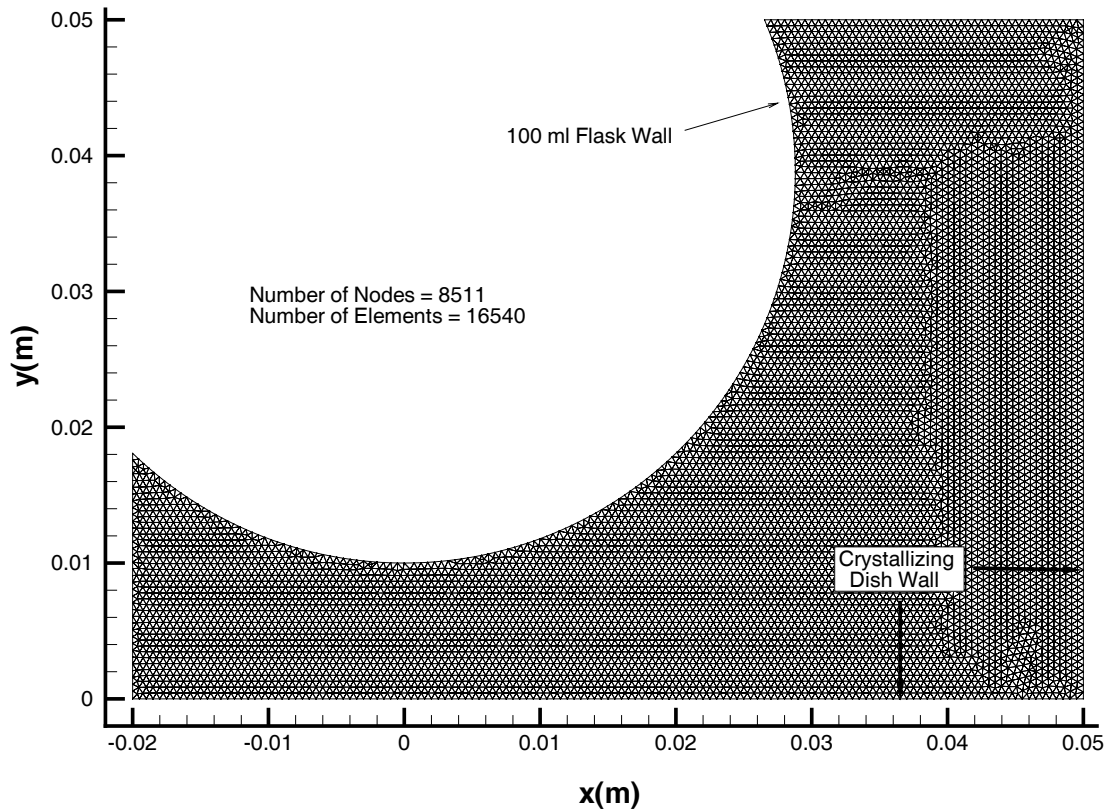


Figure 4-9 Laboratory accident simulation finite element mesh.

approximately 0.2 atm., down from an initial maximum domain pressure of 592.2 atm. at the vaporization point and an ambient domain pressure of 1 atm.

This simulation was performed with the PCICE-FEM scheme incorporating FCT as the shock capturing scheme. In order to control the nonphysical stair stepping phenomena of the FCT scheme in the regions where rarefaction wave exist, Swanson and Turkel's (1992) pressure sensor S_p of equation (3.47) was employed with the weighting factor $\alpha = 0.5$. Only a very little amount of this pressure-activated artificial dissipation is required for this simulation. The dissipation coefficient C_p of equation (2.28) was set to $C_p = 0.05$. This problem is similar to the one-dimensional shock tube simulation of Section 4.1.1 in that a fluid at rest is subjected to a sudden pressure drop. In this case, an extreme pressure drop that is initially distributed over one computational cell. Throughout simulation time, the shock waves in this simulation were captured in two to three cells. This simulation is an excellent demonstration of the FCT scheme's ability to capture transient shock waves.

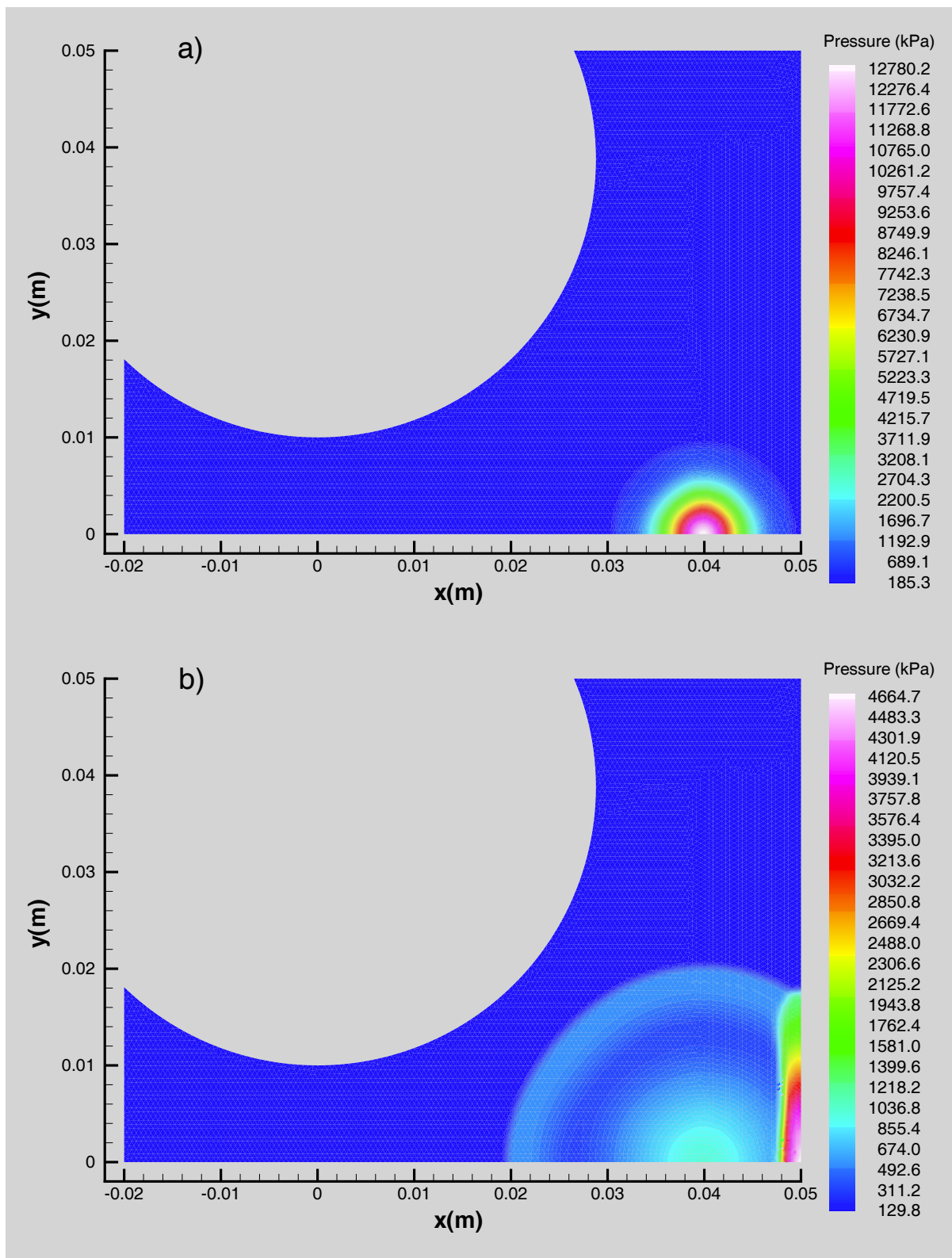


Figure 4-10 Laboratory accident simulation pressure contours at a) $t = 5\mu\text{s}$ and b) $t = 15\mu\text{s}$.

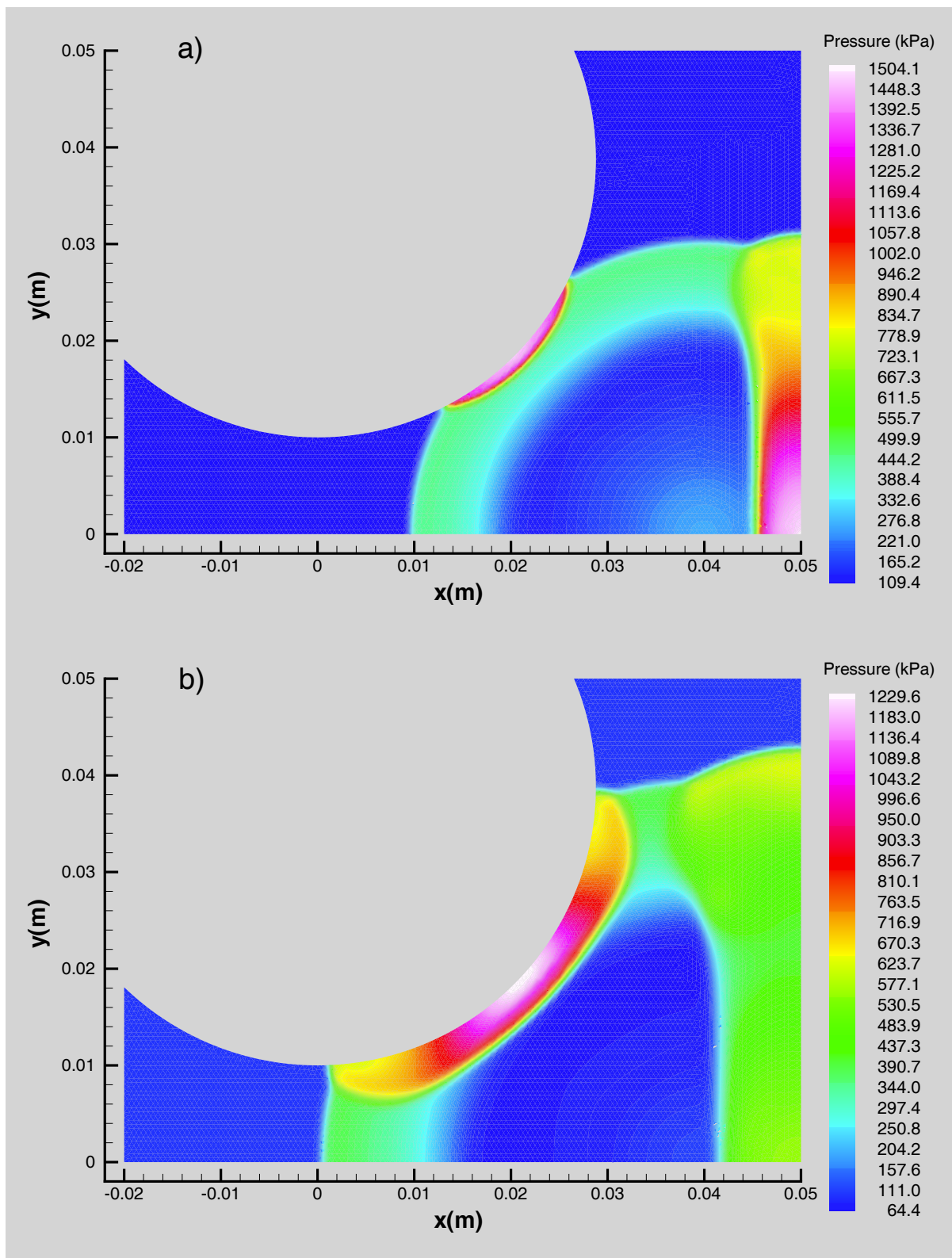


Figure 4-11 Laboratory accident simulation pressure contours at a) $t = 25 \mu s$ and b) $t = 35 \mu s$.

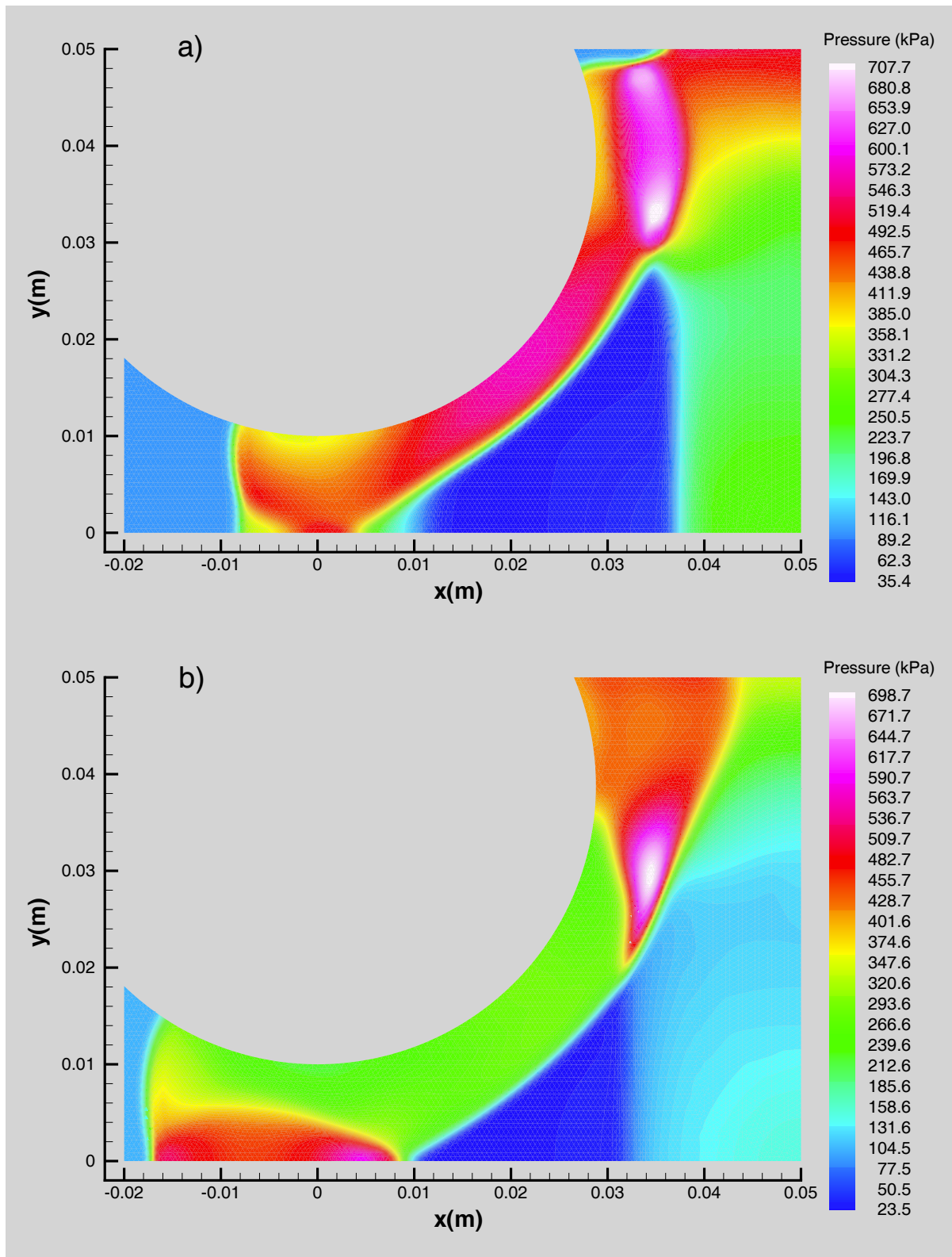


Figure 4-12 Laboratory accident simulation pressure contours at a) $t = 45 \mu s$ and b) $t = 55 \mu s$.

4.1.3 Low-Speed Viscous Flow Around a Cylinder (Von Karman Vortex Street)

Low Reynolds number flow around a cylinder in a cross flow is a common benchmark problem for transient algorithms employing the *incompressible* Navier-Stokes equations. The problem is idealized by an infinitely long cylinder placed in free stream flow normal to the axis of the cylinder. Beyond a critical Reynolds number of $Re > 40$ (e.g. White, 1991), the solution becomes unsteady and a periodic shedding of vortices occur. This oscillating phenomena is known in the literature as the von Karman vortex street. These vortices are known to be laminar for Reynolds numbers in the approximate range of $40 < Re < 5000$ (e.g. White, 1991).

Nearly incompressible flow problems, such as the von Karman vortex street simulation presented here, are difficult to solve with compressible flow algorithms incorporating the conservative form of the governing hydrodynamic equations. While significant variations in momentum may occur across the domain, density and total energy vary little in the presence of adiabatic boundary conditions. Many in the literature tend to define flow fields with a peak velocity less than a Mach number of $M < 0.3$ as an incompressible flow field. The free stream Mach number defined for this flow field, $M = 0.05$, falls well within this accepted definition of incompressible flow. The term incompressible flow is never applied to a PCICE-FEM simulation in this study because, even though the density for this simulation varies by less than 0.5% across the domain, the PCICE algorithm is formulated for variable density. The term nearly incompressible is preferred for flow fields of this type.

The computational domain for the von Karman vortex street simulation presented here is shown in Figure 4-13. The inlet boundary is located at $x = -5.0$ m and the exit boundary is located at $x = 20.0$ m. The boundaries located at $y = -7.5$ m and $y = 7.5$ m are prescribed free-slip solid walls. A 1.0 m in diameter cylinder is located at $x = y = 0.0$ m. The finite element mesh consists of 26068 nodes and 51786 triangular finite elements. Figure 4-14 is a close-up view of the boundary layer mesh generated in the vicinity of the cylinder.

The flow parameters of the von Karman vortex street simulation presented here is defined in terms of the compressible Navier-Stokes equations. The free stream velocity is equivalent to a Mach number of $M = 0.05$. At a free stream temperature and pressure of $T_\infty = 300.0$ K and $P_\infty = 101325.0$ Pa, respectively, the free stream velocity is $V_\infty = 17.36$ m/s. The Reynolds number is specified to be $Re = 100$. Expressing equation (4.1) in terms of viscous flow around a cylinder, the Reynolds number is defined (e.g. White, 1991) as

$$Re = \frac{\rho_\infty V_\infty D}{\mu_\infty}, \quad (4.6)$$

where ρ_∞ is the free stream density and D is the diameter of the cylinder. For a cylinder diameter of $D = 1.0$ m, the resulting free stream dynamic viscosity is scaled to $\mu_\infty = 0.2043$ (N·s)/m² and is applied as a constant over the entire domain. This constant viscosity assumption is valid for this simulation as the pressure and temperature distribution vary little over the domain. The stagnation pressure P_0 and temperature T_0 are computed with equations

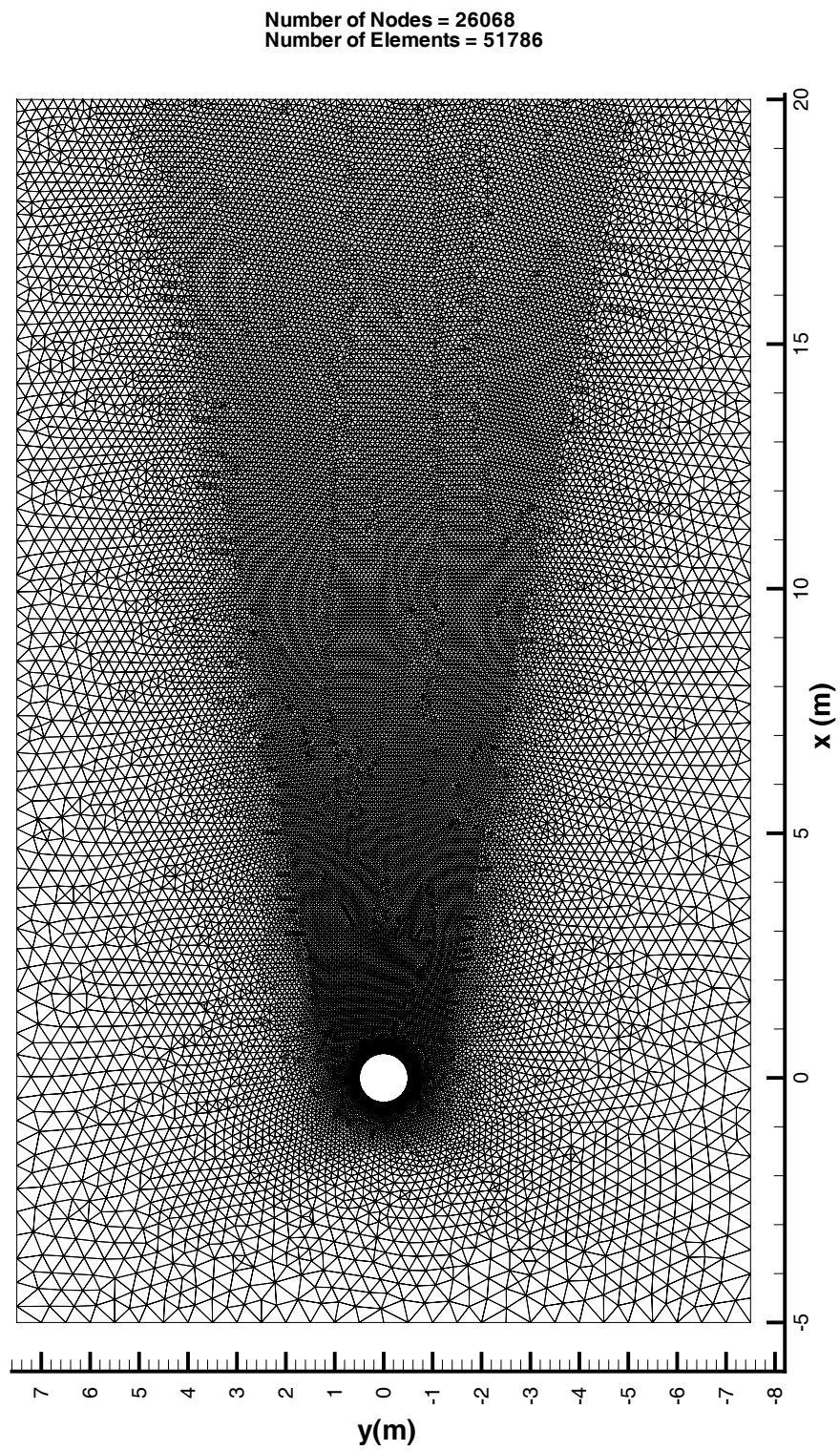


Figure 4-13 Finite element mesh for the von Karman vortex street simulation.

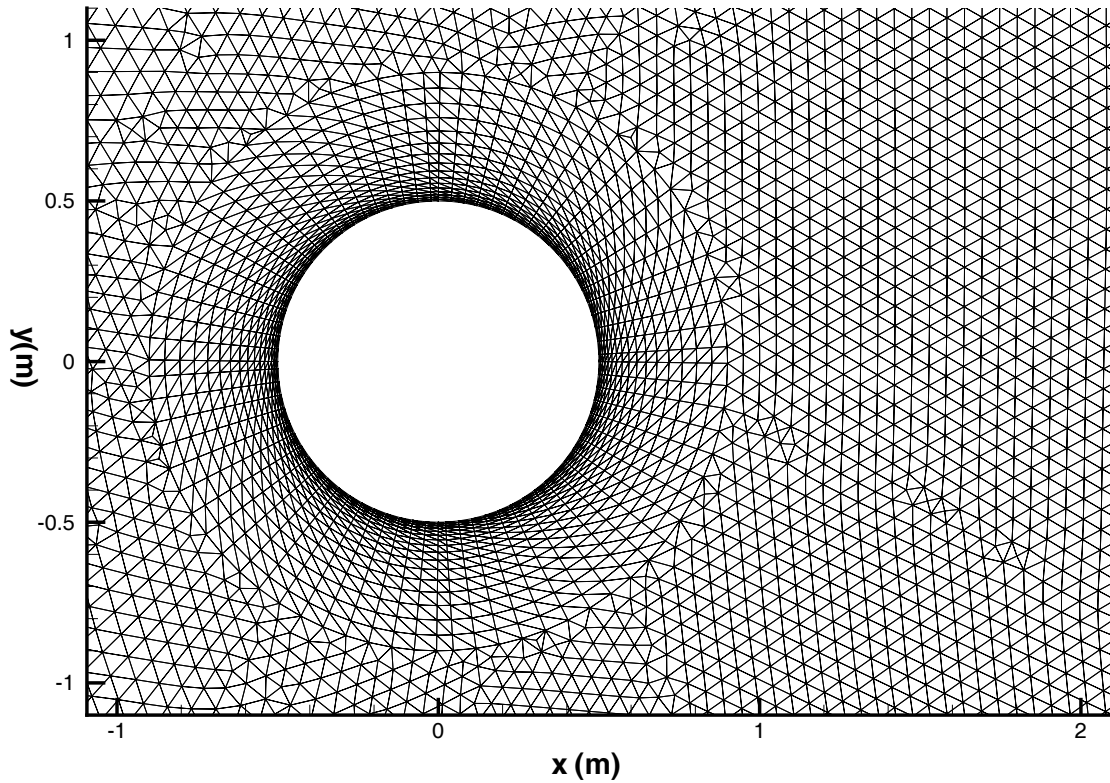


Figure 4-14 Boundary layer mesh in the vicinity of the cylinder for the von Karman vortex street simulation.

(1.7) and (1.9), respectively, and are applied to the inlet boundary with the y -component of velocity set to $v = 0.0$ m/s. The subsonic exit boundary is specified with a back pressure of $P = 101325.0$ Pa. The initial conditions for this domain at time $t^n = 0.0$ s are uniform $T = 300.0$ K, $P = 101325.0$ Pa, and $u = v = 0.0$ m/s. Adiabatic, no-slip boundary conditions are applied at the solid wall of the stationary cylinder.

With the inherent difficulty in solving nearly incompressible flows, it was found that the approximate time step size criteria of equation (3.106) could not be fully realized. The safety factor had to be reduced to $\beta = 0.5$ in order to achieve a stable time step. The addition of artificial dissipation had little effect on the upper limit of the safety factor. In fact, with such a high dynamic viscosity μ_∞ defined for this simulation, little artificial dissipation was required to stabilize the solution here. Swanson and Turkel's (1992) pressure sensor S_p of equation (3.47) was employed with the weighting factor $\alpha = 0.95$, which favors Peraire's artificial dissipation. The dissipation coefficient C_p of equation (2.28) was set to $C_p = 0.25$. The pressure sensor S_p had the most effect at the beginning of the simulation when the largest pressure wave propagated across the domain. After the stable oscillatory profile was achieved, the pressure variations are so small that the addition of artificial dissipation were negligible.

Figure 4-15 through Figure 4-18 are the flooded contour representations of the solution to the von Karmen vortex street problem at a time of $t = 4.0$ s for pressure, density, temperature, and Mach number, respectively. The periodic oscillations in the solution of this problem are clearly visible for each contoured variable. The pressure solution represented in Figure 4-15 clearly shows the low-pressure center of each vortex propagating downstream from the stationary cylinder. The PCICE-FEM pressure solution slightly over-predicts the stagnation pressure at the stagnation point in front of the cylinder by approximately 19 Pa due to very slight numerical oscillations in the flow variables in this region. This is possibly due to the negligible amount of artificial dissipation applied in this region. Figure 4-16 represents the PCICE-FEM solution for density. The density variation across the domain is approximately 0.4%, yet the variations are clearly resolved. The temperature solution illustrated in Figure 4-17 is the most dramatic representation of the flow field. The maximum variation in temperature across the domain is on the order of 0.25 K. With negligible compression, this small variation in temperature must be due to viscous heating. Note that the track of the fluid is at a higher temperature than the surrounding ambient fluid temperature of 300.0 K. Again, there is a slight over-prediction of a stagnation value. The stagnation point in front of the cylinder is 0.014 K over the specified stagnation temperature of $T_0 = 300.15$ K. Figure 4-18 represents the Mach number solution. Mach number for this simulation is proportional to velocity magnitude because the sound speed across the domain is nearly constant. The velocity vectors in the vicinity of the cylinder are shown in Figure 4-19. The vortex shedding phenomena is clearly visible.

4.2 Steady-State Flow Simulations

The three steady-state simulations presented here are designed to demonstrate the PCICE-FEM scheme's ability to accurately converge to a unique solution for a wide range of flow regimes. The first of three steady-state simulations presented here is of an inviscid supersonic flow in a channel with an 18 degree wedge-shaped obstruction that results a complex shock wave reflection pattern. The second steady-state simulation is a viscous (laminar) double-throated converging-diverging nozzle with flows varying from Mach = 0.01 to Mach = 2.7. The final steady-state simulation is of thermally driven flow in a square cavity for three different Rayleigh numbers.

4.2.1 Mach Reflection in a Channel from an 18 Degree Wedge Obstruction

Consider a supersonic channel with two solid walls, the top wall being straight and parallel to the inlet flow direction and with a portion of the bottom wall inclined toward the inlet flow. If the inlet flow velocity is high enough and the bottom surface is not overly inclined, an oblique shock attached to the beginning of the inclined surface will form. If the angle of the shock wave is sufficiently large, a normal shock wave called a Mach reflection emanating in a direction normal to the straight top wall will form. Across the normal shock wave, the flow transitions from supersonic to subsonic flow. At the intersection of the oblique and normal shock waves, a slip line (similar to a contact discontinuity) and a reflected shock wave are formed. This intersection point is commonly referred to as a triple point. Pressure across the

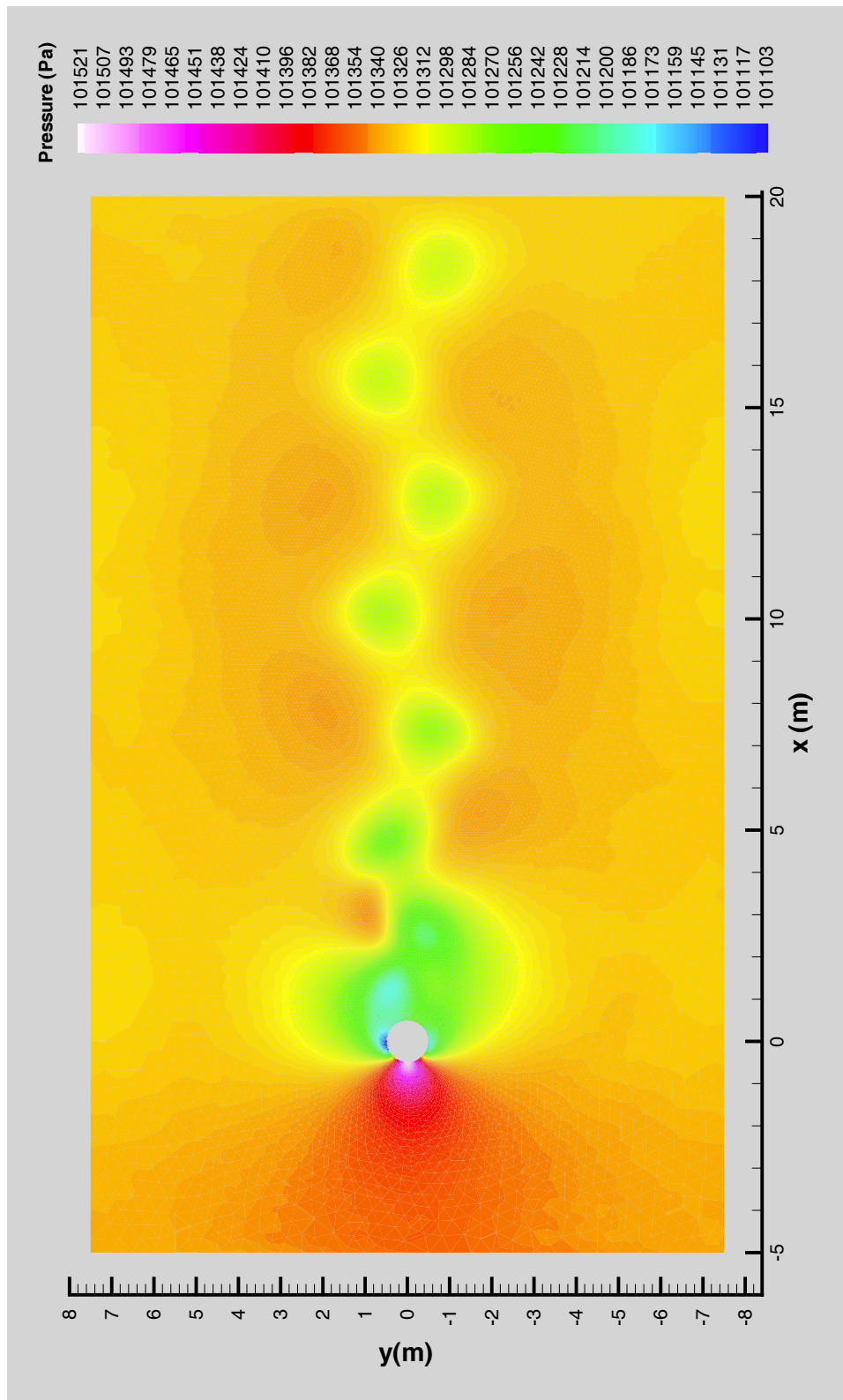


Figure 4-15 Von Karman vortex street pressure solution at time $t = 4.0$ s.

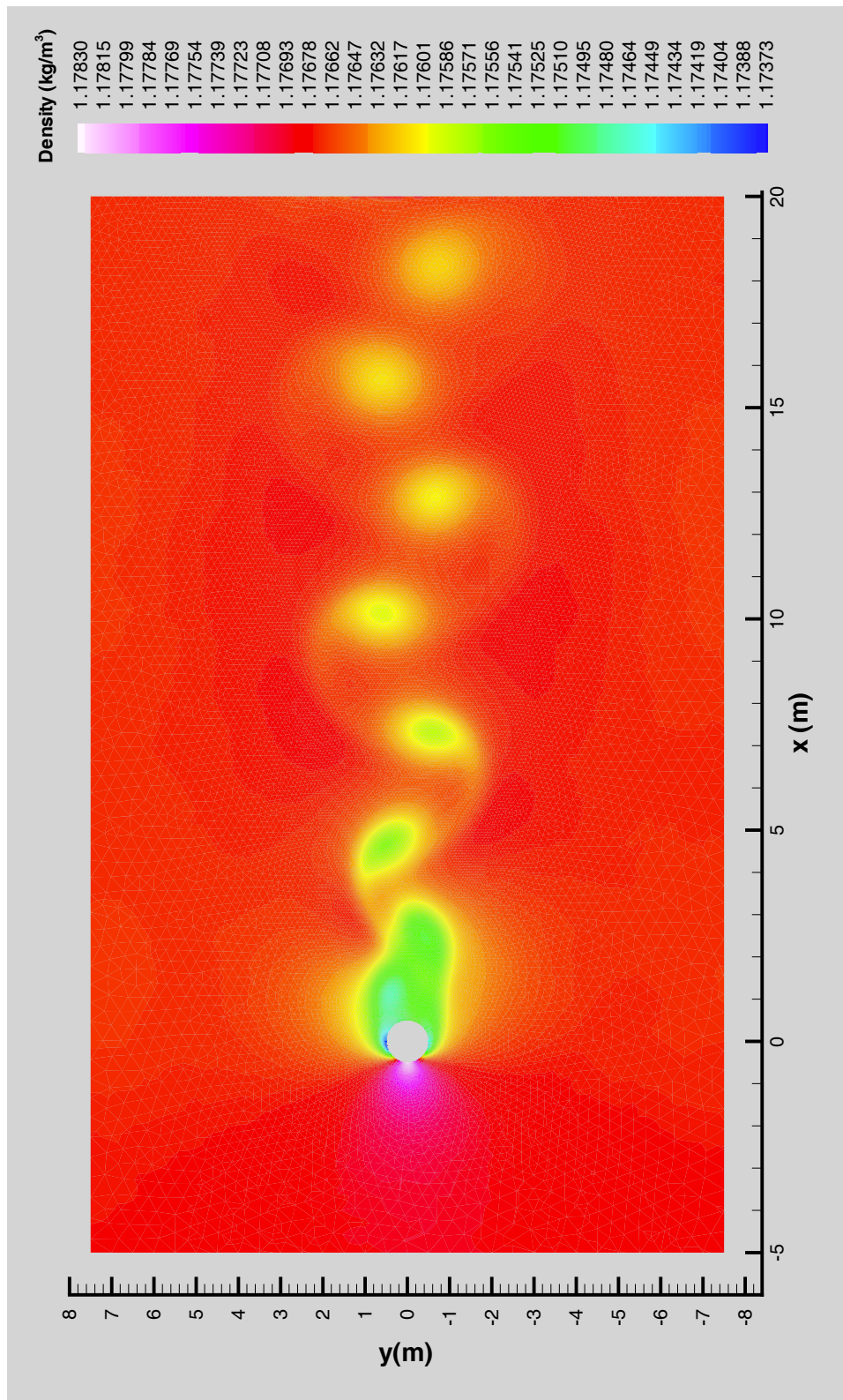


Figure 4-16 Von Karman vortex street density solution at time $t = 4.0$ s.

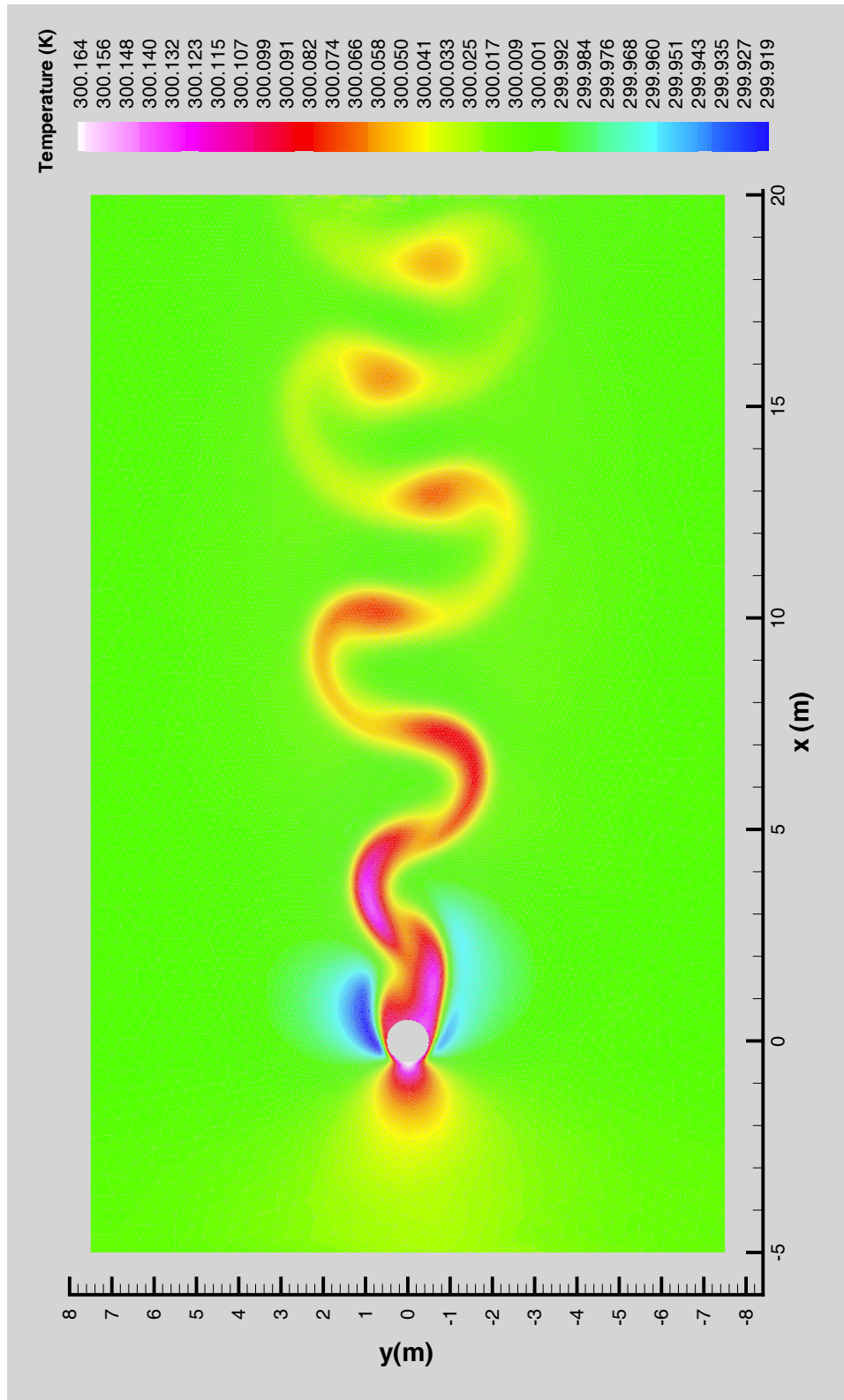


Figure 4-17 Von Karman vortex street temperature solution at time $t = 4.0$ s.

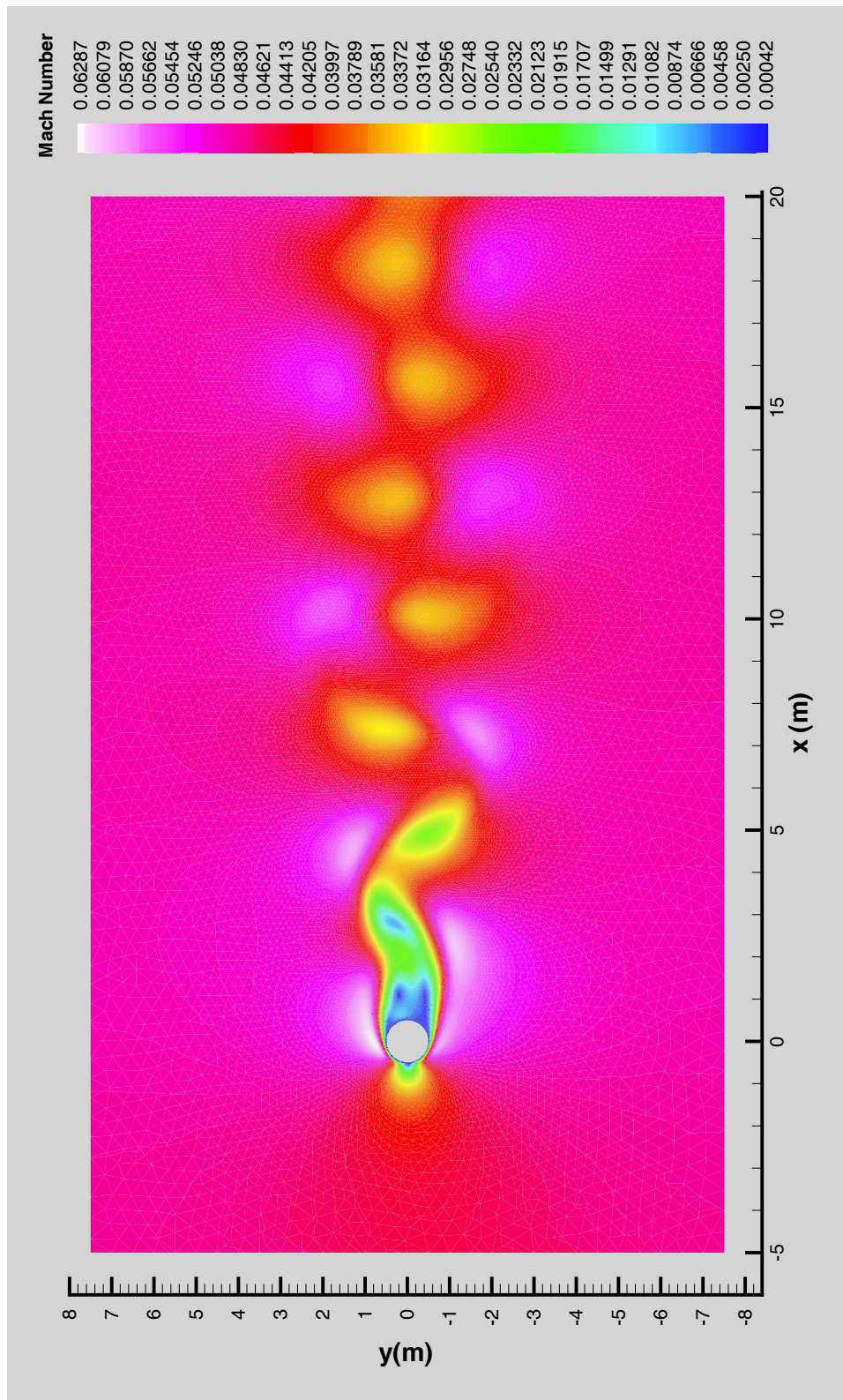


Figure 4-18 Von Karman vortex street Mach number solution at time $t = 4.0$ s.

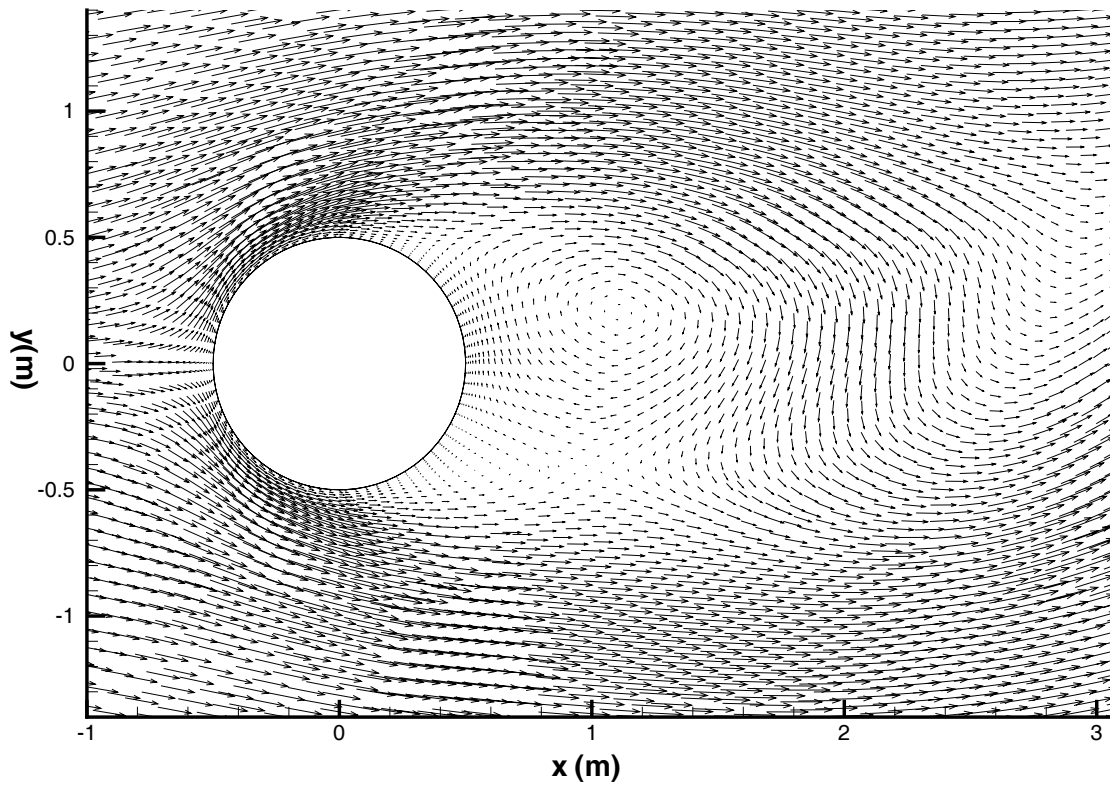


Figure 4-19 Velocity vectors in the region of the stationary cylinder for the von Karman vortex street at time $t = 4.0$ s.

slip line is continuous while density, temperature, and velocity experience a discontinuity across the slip line.

The Mach reflection simulation provided here is taken from Saad's (1985) text on compressible flow. It consists of a supersonic channel with an inlet Mach number of $M = 2.2$ and 18 degree inclined surface toward the inlet flow. The top surface is parallel to the inlet flow direction. This simulation is used here to demonstrate the PCICE-FEM scheme's ability to accurately resolve steady-state supersonic flow phenomena by comparing Saad's solution, obtained with the governing shock wave relations, to the numerical solution obtained from the PCICE-FEM scheme with FCT employed as the shock capturing scheme.

The problem geometry and finite element mesh for this Mach reflection simulation is shown in Figure 4-20. The mesh is the result of three h -refinements (Löhner, 1985a) with a normalized Laplacian of density as the error indicator. The result is an adapted mesh based upon the curvature of the density distribution. Saad's solution applies only to the region defined by $0.5 \leq x \leq 1.5$. Length units of the geometry are redundant as the problem is inviscid and steady-state. The region of the mesh extending from $x = 0.0$ to $x = 0.5$ exists to avoid shock wave interaction with the inlet boundary and the inclined surface. The region of the

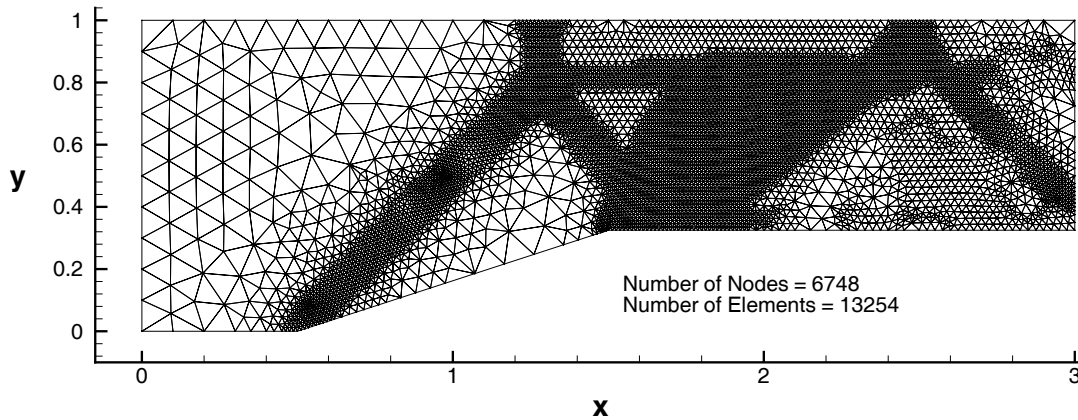


Figure 4-20 Adapted finite element mesh for inviscid Mach reflection simulation.

mesh extending from $x = 1.5$ to $x = 3.0$ is designed to investigate how much the slip line degrades from interaction with reflected shock waves and artificial viscosity. The inlet conditions of $P = 101325.0$ Pa, $T = 300.0$ K, $u = 763.87$ m/s, and $v = 0.0$ m/s are applied at the inlet surface located at $x = 0.0$. The exit surface, located at $x = 3.0$, is left free (non-reflecting). All other surfaces are specified to be free-slip solid walls. The initial conditions are equal to the inlet conditions and were applied uniformly across the domain.

The PCICE-FEM solution with FCT presented below was obtained with the following simulation parameters. The semi-implicit time weighting factor was set to $\phi = 0.75$ and the residual smoothing coefficient of equation (2.18) was set to $\varepsilon = -0.1$. While these simulation parameters are problem dependent, the values used here are typical for the steady-state solutions of supersonic inviscid domains with the PCICE-FEM scheme. With residual smoothing employed, the safety factor on the time step size criteria, equation (3.105), was set to $\beta = 1.0$. In addition to FCT, a small amount of artificial dissipation was applied by equation (2.28). Swanson and Turkel's (1992) pressure sensor S_p of equation (3.47) was employed with the weighting factor $\alpha = 0.5$, which equally weights Paire's and Jameson's artificial dissipation methods. The dissipation coefficient C_p of equation (2.28) was set to $C_p = 0.05$.

Figure 4-21 through Figure 4-24 are the PCICE-FEM steady-state solution with FCT for this Mach reflection simulation. The triple point is clearly defined in the pressure solution presented in Figure 4-21. The oblique shock exists in the region extending from $x = 0.5$ to $x \approx 1.25$ and from $y = 0.0$ to $y \approx 0.75$. The normal shock exists at $x \approx 1.25$ and extends from $y \approx 0.75$ to $y = 1.0$. From the intersection point of the oblique shock and the normal shock (the location of the triple point), the reflected shock emanates down and to the right. This shock creates two more weak reflections before exiting the domain. Saad gives the ratio between the pressure behind the oblique shock to the inlet pressure as 2.67. the pressure solution given by

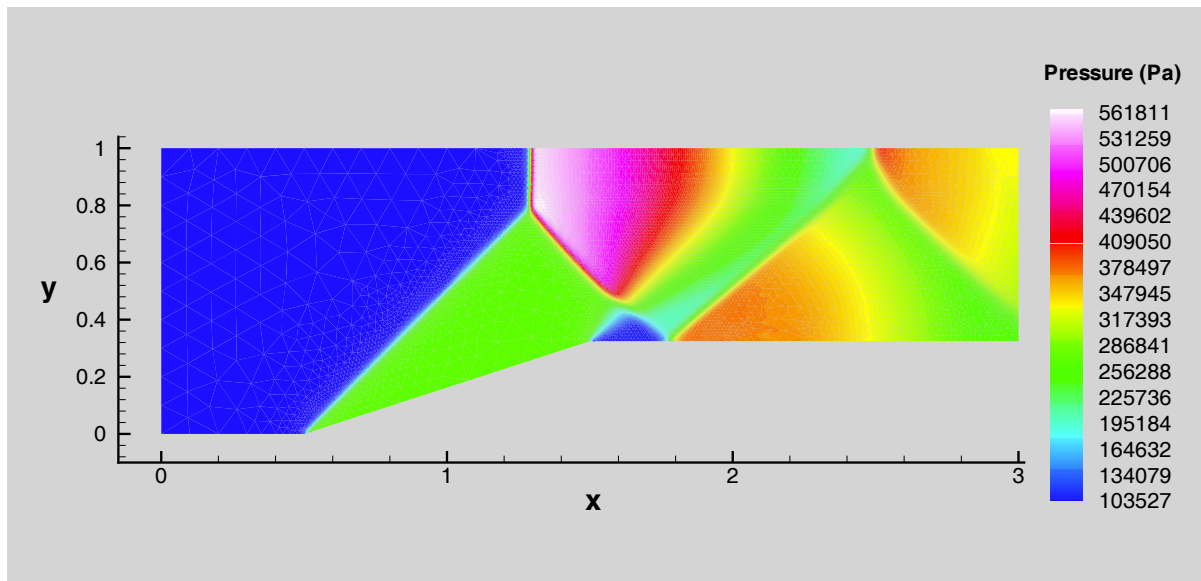


Figure 4-21 PCICE-FEM Mach reflection solution for pressure.

the PCICE-FEM scheme is well inside the precision of this value. Saad iteratively found the pressure ratio between the pressure located directly behind the normal shock to the inlet pressure as 5.5. Again, the PCICE-FEM pressure solution agrees.

The density, temperature, and Mach number PCICE-FEM solutions are presented in Figure 4-22 through Figure 4-24, respectively. In these figures, the slip line emanating from the triple

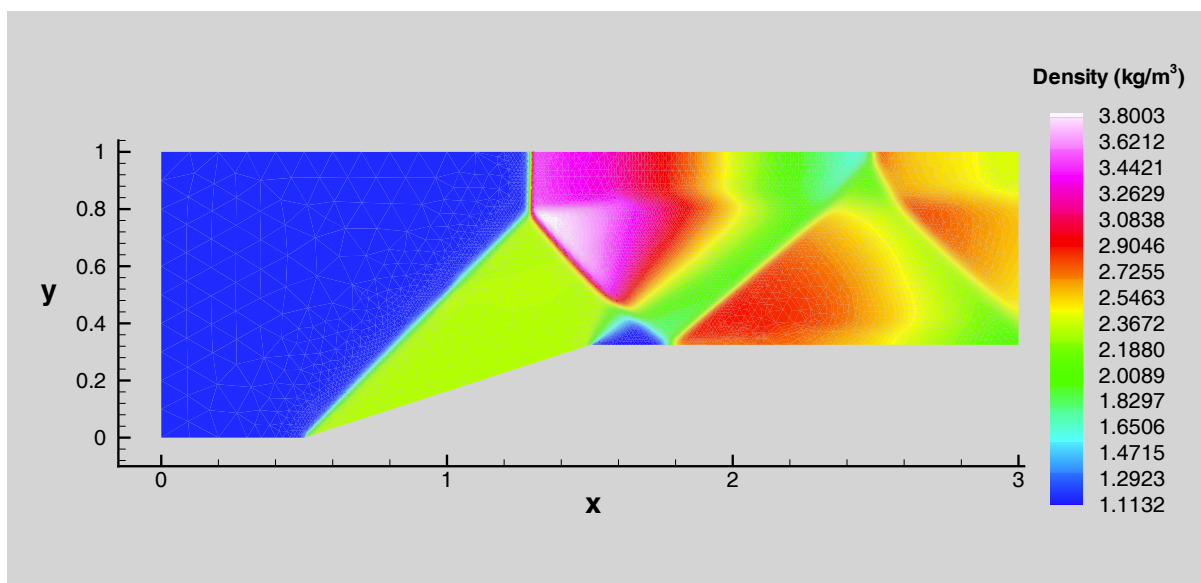


Figure 4-22 PCICE-FEM Mach reflection solution for density.

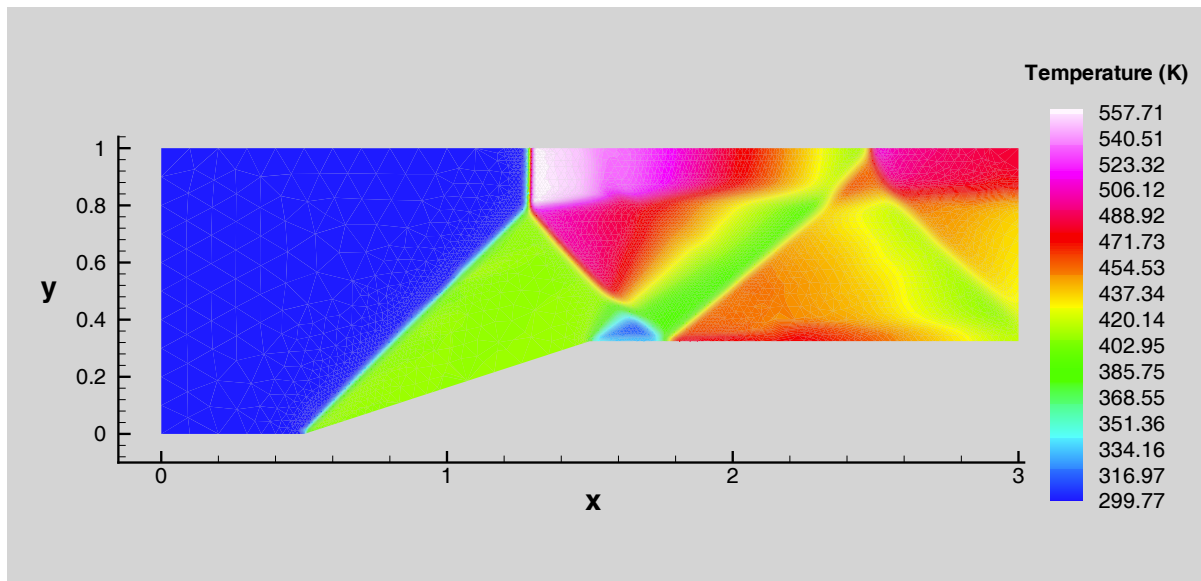


Figure 4-23 PCICE-FEM Mach reflection solution for temperature.

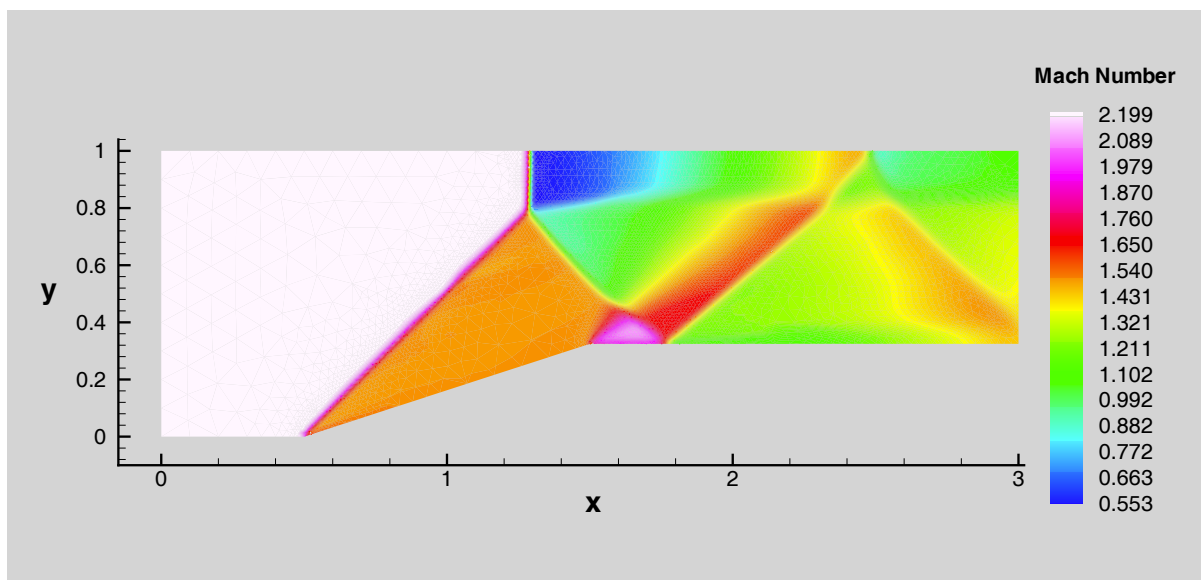


Figure 4-24 PCICE-FEM Mach reflection solution for Mach number.

point is clearly defined. The slip line is noticeably diffused after the third shock reflection. As with pressure, the PCICE-FEM Mach number solution agrees very well with Saad's solution. In Figure 4-24, the lowest Mach number in the domain is directly behind the normal shock. The contour legend gives the lowest value of Mach number as $M = 0.553$. Saad's iterative value is $M = 0.55$. In the constant region behind the oblique shock, Saad's solution gives the Mach number as $M = 1.48$. The PCICE-FEM Mach number solution in this region is a nearly constant value of $M = 1.49$. Note that Saad estimated his value from the shock tables.

4.2.2 Double-Throated Nozzle (The GAMM Problem)

The viscous (laminar) double-throated nozzle simulation presented in this section is the result of the 1987 GAMM workshop (Bristeau, et al.,1987). This problem was the second of two test cases designed with the aim of generating strong viscous interaction phenomena in steady-state, laminar compressible flows inside a well-bounded domain. Supersonic flow conditions are obtained in the first converging-diverging nozzle. then the wall is turned concave toward the second converging-diverging nozzle. It is in this middle diverging-converging section, with partly supersonic flow conditions, that compression waves, oblique shock waves, and boundary layer separations are expected to occur. After the second throat, the flow is allowed to expand rapidly in the second diverging section.

This problem was chosen as a steady-state, viscous test of the PCICE-FEM scheme for the following reasons: 1) The geometry and operating conditions for this problem require a large variation in element size and flow velocity resulting in a stiff numerical integration problem. The flow velocity varies from a Mach number of $M = 0.01$ at the inlet to $M = 2.7$ at the exit. The geometry dictates that relatively small element volumes will be required to accurately resolve the flow features between the nozzle throats. 2) It is a high-speed, laminar flow test case and, thus, does not require a turbulence model which has not yet been incorporated into the PCICE-FEM code constructed for this research effort. 3) There are numerous sets of computational results documented in the reference (Bristeau, et al.,1987) with which to compare.

The problem geometry and finite element mesh for the double-throated nozzle problem presented here is shown in Figure 4-25. Only half of the mesh shown was used for the computation as there is a symmetry plane defined at $y = 0.0$ m. The solid wall is defined by a complex set of clamped cubic splines (see Bristeau, et al.,1987). Dense clustering of the linear triangular finite elements near the solid wall was performed in order to resolve the boundary layer phenomena. This adapted mesh is the result of a single h -refinement (Löhner, 1985a) based upon variations in density. The inlet is located at $x = 0.0$ m and the exit is defined by the surfaces for $x > 14.0$ m.

The operating parameters for this simulation are defined by an inlet reservoir and an isothermal solid wall. Reservoir conditions, or stagnation conditions, applied at the inlet were chosen to be $P_0 = 253312.5$ Pa and $T_0 = 400.0$ K with the y -component of velocity suppressed. The solid wall is defined as no-slip and isothermal with the wall temperature set to the stagnation temperature, $T_w = T_0$. The exit boundary surfaces are supersonic, thus they are left free and nothing is specified. The Reynolds number for this problem is set to $Re = 1600$, where the Reynolds number is defined (Bristeau, et al.,1987) as

$$Re = \frac{\rho_0 a_0 L}{\mu_0}. \quad (4.7)$$

For this Reynolds number, L is the nozzle half-height located at the first throat, $x = 0.0$ m, and is chosen here to be $L = 1.0$ m. In equation (4.7), a_0 and ρ_0 are the stagnation sound speed and density. These parameters define a constant dynamic viscosity value of $\mu_0 = 0.5528$ (N·s)/m².

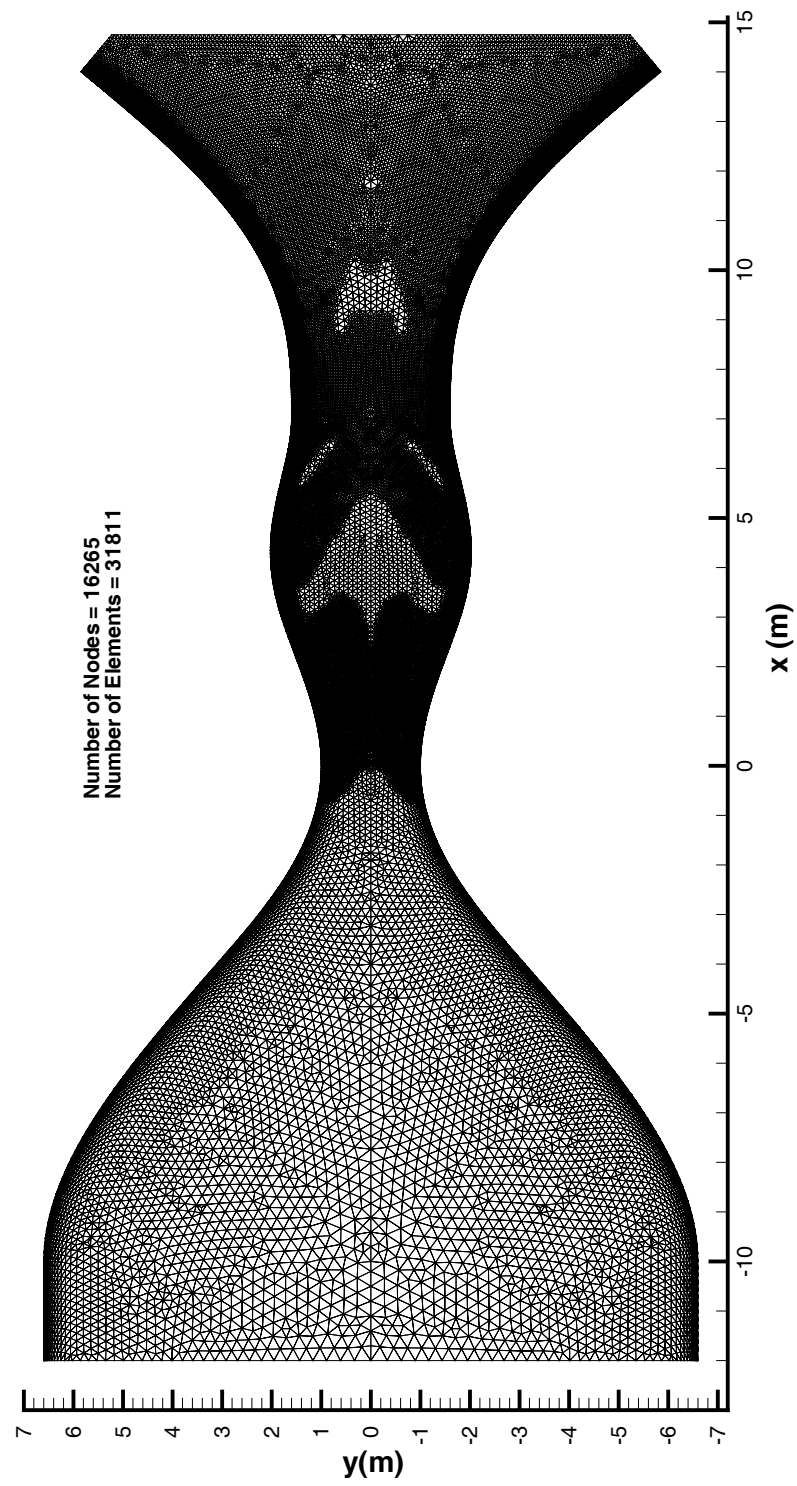


Figure 4-25 Adapted finite element mesh for the double-throated nozzle simulation.

The initial conditions were designed to guarantee supersonic flow at the exit. Shock tube type initial conditions are applied with $P = P_0$ and $T = T_0$ for $x < 0.0$ m and $P = 25000.0$ Pa and $T = 400.0$ K for $x \geq 0.0$ m. Static velocity conditions are initially applied across the domain.

The PCICE-FEM solution of the double-throated nozzle simulation is depicted in Figure 4-26 through Figure 4-29 for pressure, density, temperature, and Mach number, respectively. The solution incorporated FCT as the high-resolution filter for shock capturing. Due to the extremely viscous nature required to maintain laminar flow, no other artificial dissipation method was incorporated. The PCICE-FEM solution compares relatively well with the results documented in the GAMM workshop (Bristeau, et al., 1987). However, there was one exception. There are two boundary layer separation/re-attachment regions, or recirculation bubbles. The first is located just downstream from the first throat. This bubble is caused by an adverse pressure gradient in the first diverging section. The second recirculation bubble appears just downstream of the second throat and is due to the interaction between the impinging oblique shock wave and the viscous boundary layer. The PCICE-FEM solution generates these recirculation bubbles somewhat farther upstream than those in the documented workshop results. This is believed to be due to the much finer mesh employed in the recirculation regions for this simulation than those employed by the researchers participating in the GAMM workshop. Also, FCT minimizes artificial diffusion and provides for much sharper shock profiles than the artificial dissipation methods employed in the workshop.

Comparison of the PCICE-FEM results for the double-throated nozzle simulation with obtained in the GAMM workshop is somewhat unfair to the researchers participating in the workshop. The fastest available computer for the workshop was a CRAY-1s. While this was a state of the art machine at the time of the workshop, a modern desktop personal computer is considerably faster. The 1 GHz AMD K7 processor used in this study has an average computational speed approximately 2.5 times faster than the CRAY-1s. Also, the CRAY-1s had only 8 Mb of memory versus the 256 Mb available here. The GAMM researchers probably had difficulty with memory storage and had to employ the hard drive swap space, which would greatly increase run times. With the PCICE-FEM code employed here, total runtime was less than three hours. It took approximately 19,000 time steps and 45 minutes for the initial mesh and approximately 12,000 time steps and 2 hours for the adapted mesh to achieve the results illustrated in Figure 4-26 through Figure 4-29. The time-weighting factor was set to $\phi = 0.67$ and the residual smoothing coefficient was set to $\varepsilon = -0.1$.

4.2.3 Thermally Driven Flow in a Square Cavity

The thermally driven flow in a square cavity problem being considered here is described by a two-dimensional circulatory flow of air (Prandtl number $Pr = 0.72$) in an upright square domain of side length L . The horizontal walls are defined by no-slip, adiabatic solid walls and the vertical walls are defined by no-slip, isothermal solid walls. The two vertical walls are assigned different temperatures. This natural convection problem was the topic of invited contributions at the 1981 Venice conference on Numerical Methods in Thermal Problems. The results from this conference were reported in detail by Jones and Thompson (1981) and

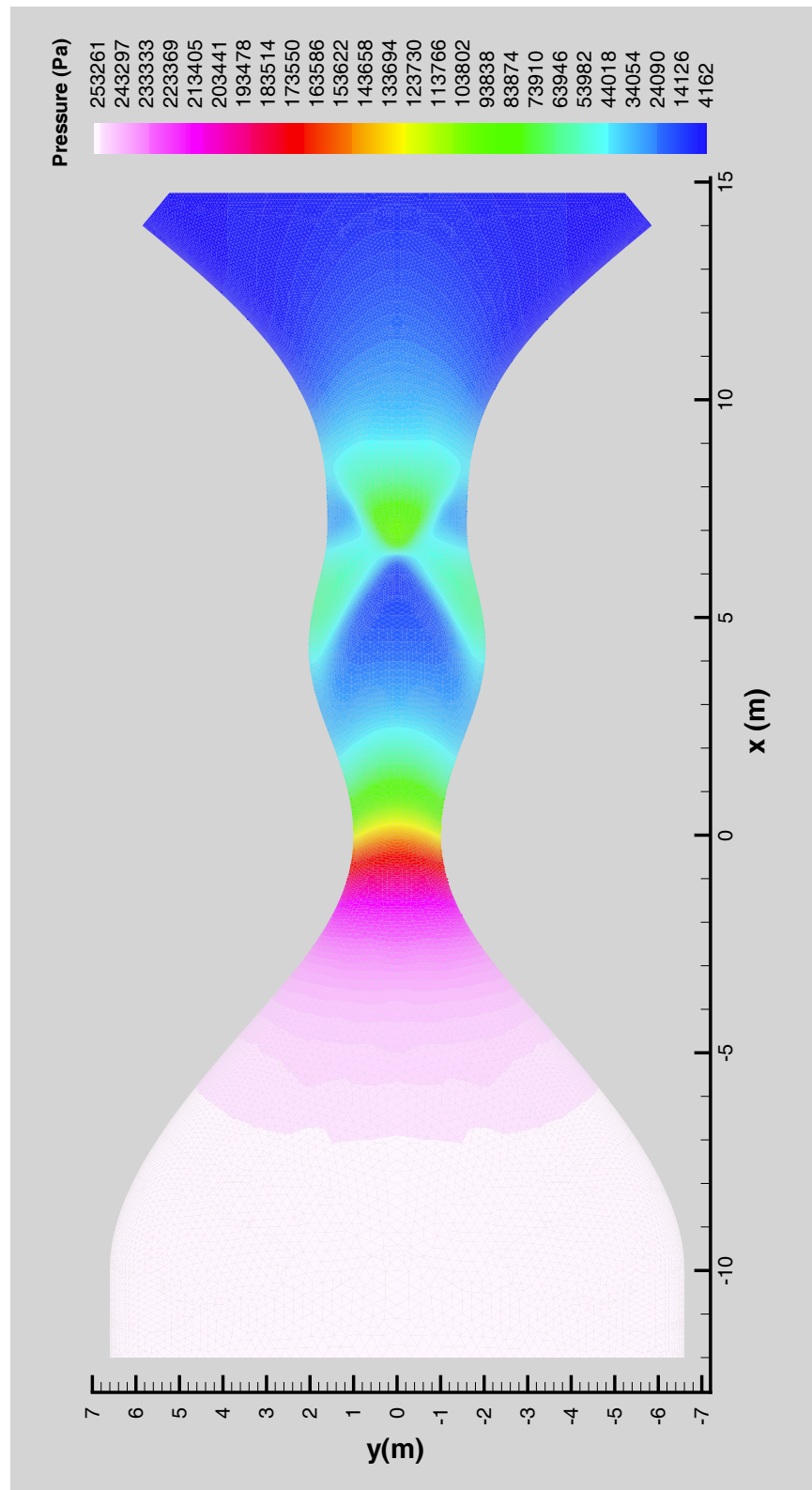


Figure 4-26 PCICE-FEM pressure solution for the double-throated nozzle simulation.

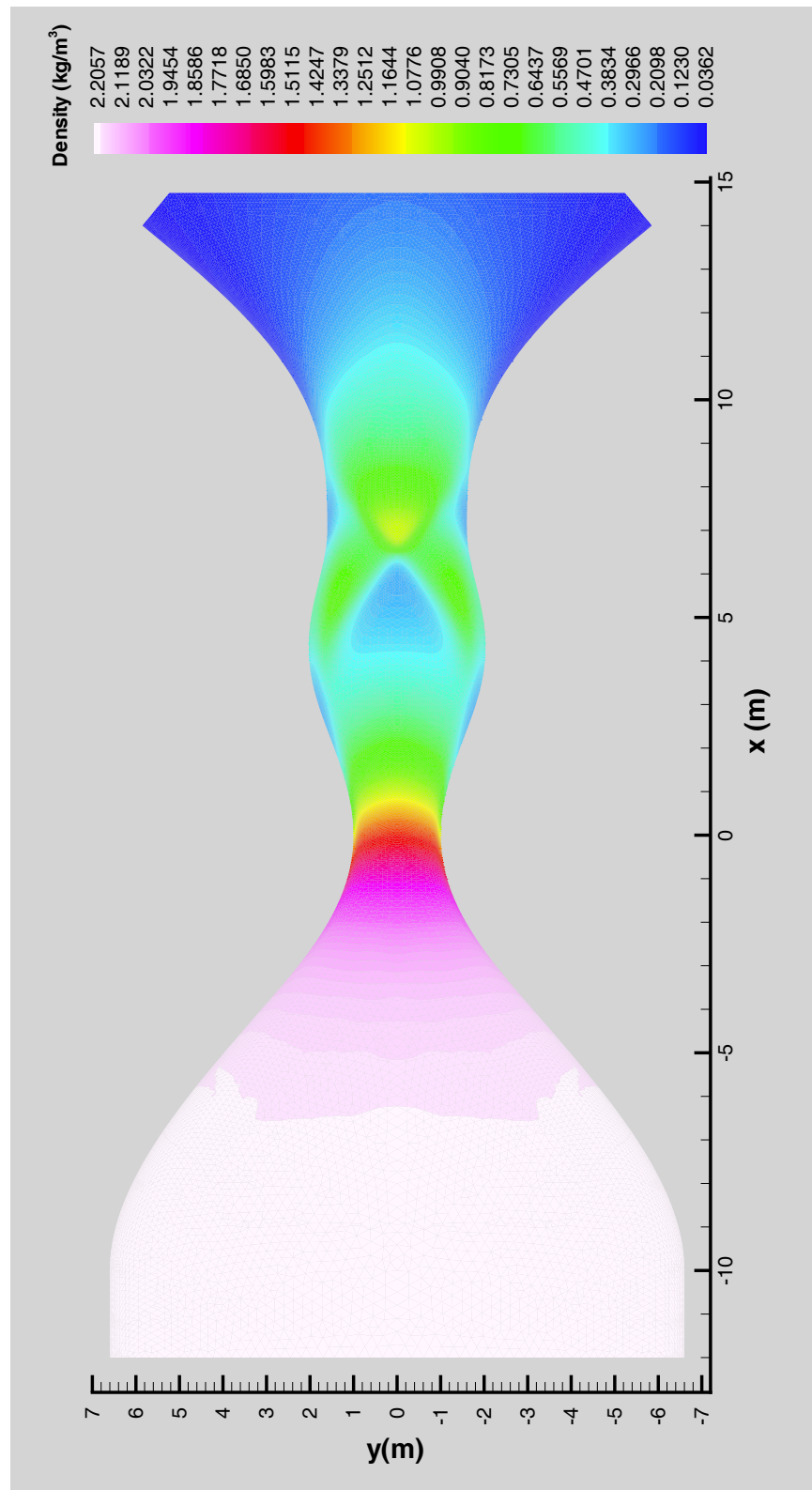


Figure 4-27 PCICE-FEM density solution for the double-throated nozzle simulation.

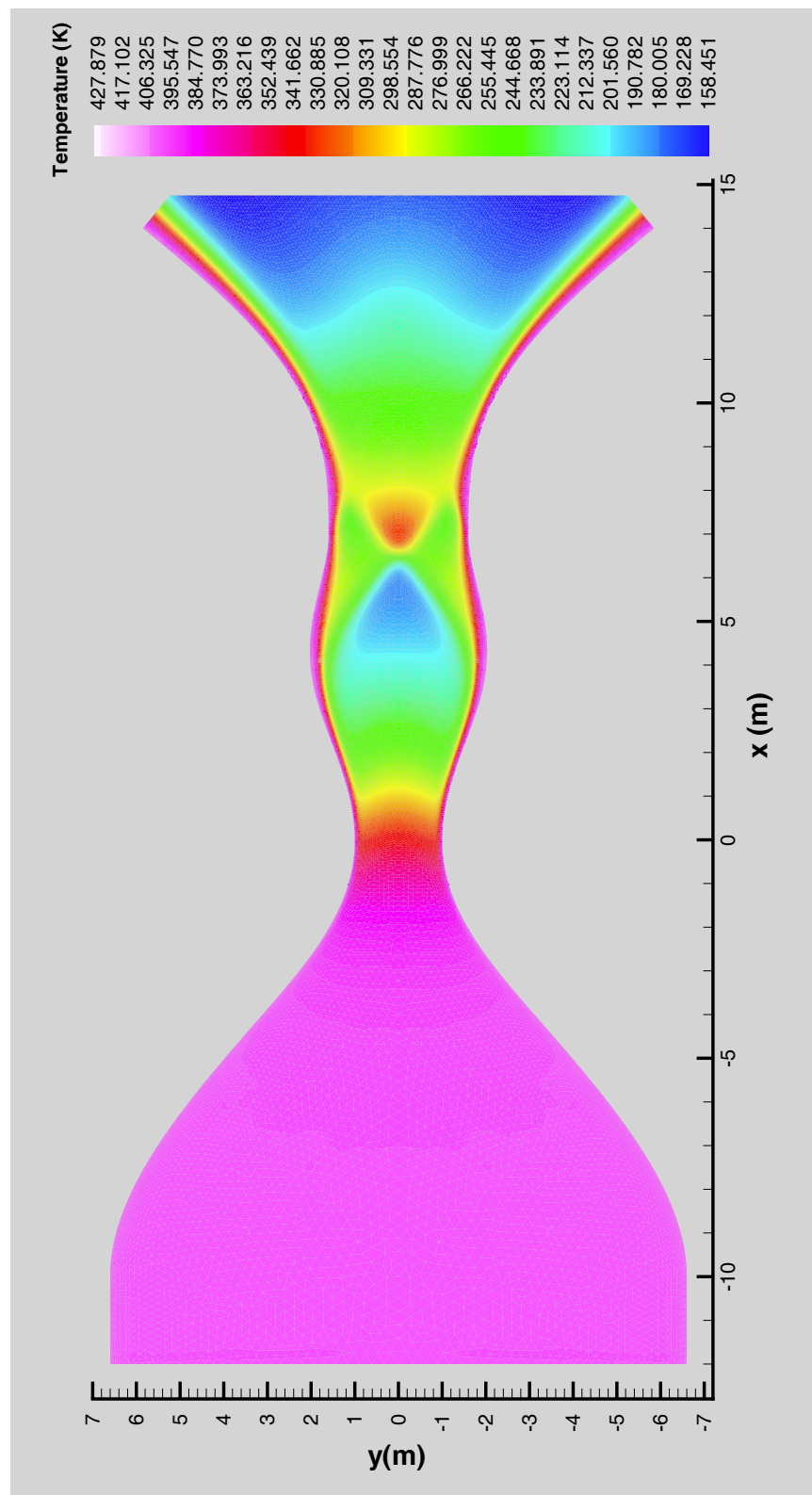


Figure 4-28 PCICE-FEM temperature solution for the double-throated nozzle simulation.

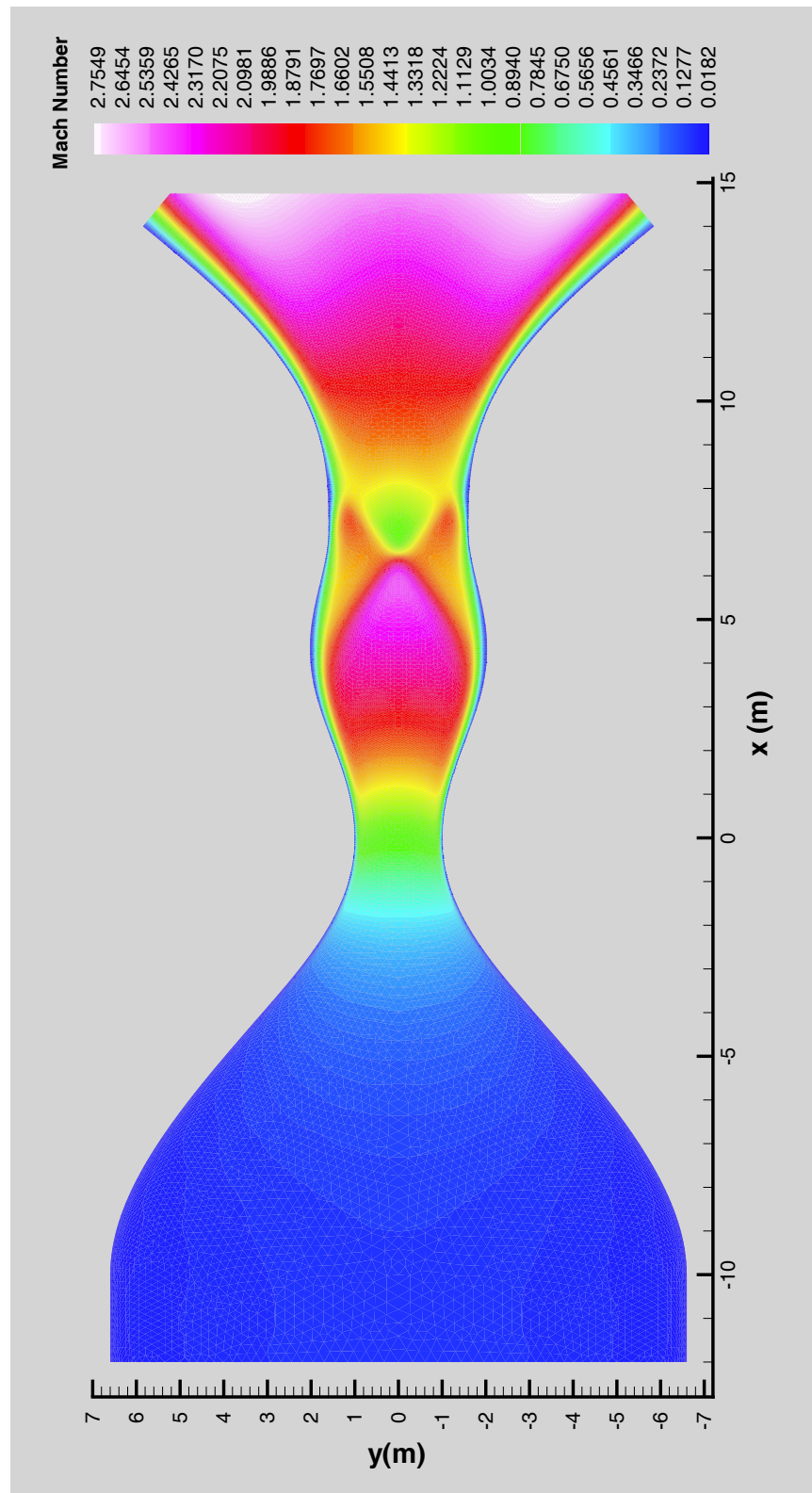


Figure 4-29 PCICE-FEM Mach number solution for the double-throated nozzle simulation.

summarized in Davis (1983) and Davis and Jones (1983). The governing hydrodynamic equations prescribed by this conference were the incompressible Navier-Stokes equations with the Boussinesq approximation. The temperature solution was governed by a primitive variable form of the thermal energy equation.

It was initially hoped that, by simulating the thermally driven flow in this square cavity problem, the results reported in Davis (1983) would validate the PCICE-FEM scheme as the flow approached the incompressible limit with no Boussinesq approximation. However, the solution results obtained here are markedly different than those reported by Davis. Unfortunately, this problem is difficult to reproduce in the laboratory to verify numerical results because of end effects produced normal to the two-dimensional plane (three-dimensional effect). In order to document the differences in the solution results between a constant density (incompressible) formulation with the Boussinesq approximation and the variable density formulation of the PCICE-FEM scheme, three simulations will be performed here at Rayleigh numbers of $Ra = 1.0 \times 10^5$, 5.0×10^5 , and 1.0×10^6 .

The triangular finite element mesh for this thermally driven problem is presented in Figure 4-30. This mesh is constructed from a 121×121 structured grid resulting in 14641 nodes and 28800 triangular finite elements. Dense clustering of the nodes near the solid walls was performed to accurately resolve the viscous and thermal boundary layers. The side length dimension of the cavity was conveniently chosen to be $L = 1.0$ m. The prescribed thermal boundary conditions for this problem are defined in the figure. The differential temperature between the two vertical walls was chosen as 20.0 K to keep the fluid velocities low. As will be discussed later, gravity and temperature are the key driving mechanisms (versus pressure) of the flow in this problem and is specified as $g = 9.81 \text{ m/s}^2$.

A body force term, to account for the effects of gravity, is needed for this buoyancy driven problem. By making use of the elemental time-weighted density solution from the first step of the two-step Taylor-Galerkin, the momentum source term for the y-component of the semi-implicit balance of momentum, equation (3.7), is defined as

$$d_y^{n+\varphi} = -9.81 \rho_e^{n+\varphi} . \quad (4.8)$$

High-order temporal accuracy is not important for this steady-state problem. But because elemental values are needed for the source terms, the time-weighted elemental value of density from the first step is convenient to use.

The values of dynamic viscosity for this series of simulations are determined from the initial conditions and the thermal boundary conditions. The initial conditions for these simulations are static velocity and a uniform pressure, $P = 101325.0$ Pa, and temperature, $T = 300.0$ K. From these initial conditions, the initial density is $\rho = 1.17666 \text{ kg/m}^3$. The coefficient of thermal expansion may then be computed with equation (4.4) to give $b = 3.333 \times 10^{-3} \text{ 1/K}$. With $\Delta T = 20.0$ K from the thermal boundary conditions, the values of the dynamic viscosity are then computed from the definition of Rayleigh number, equation (4.2). For Rayleigh numbers

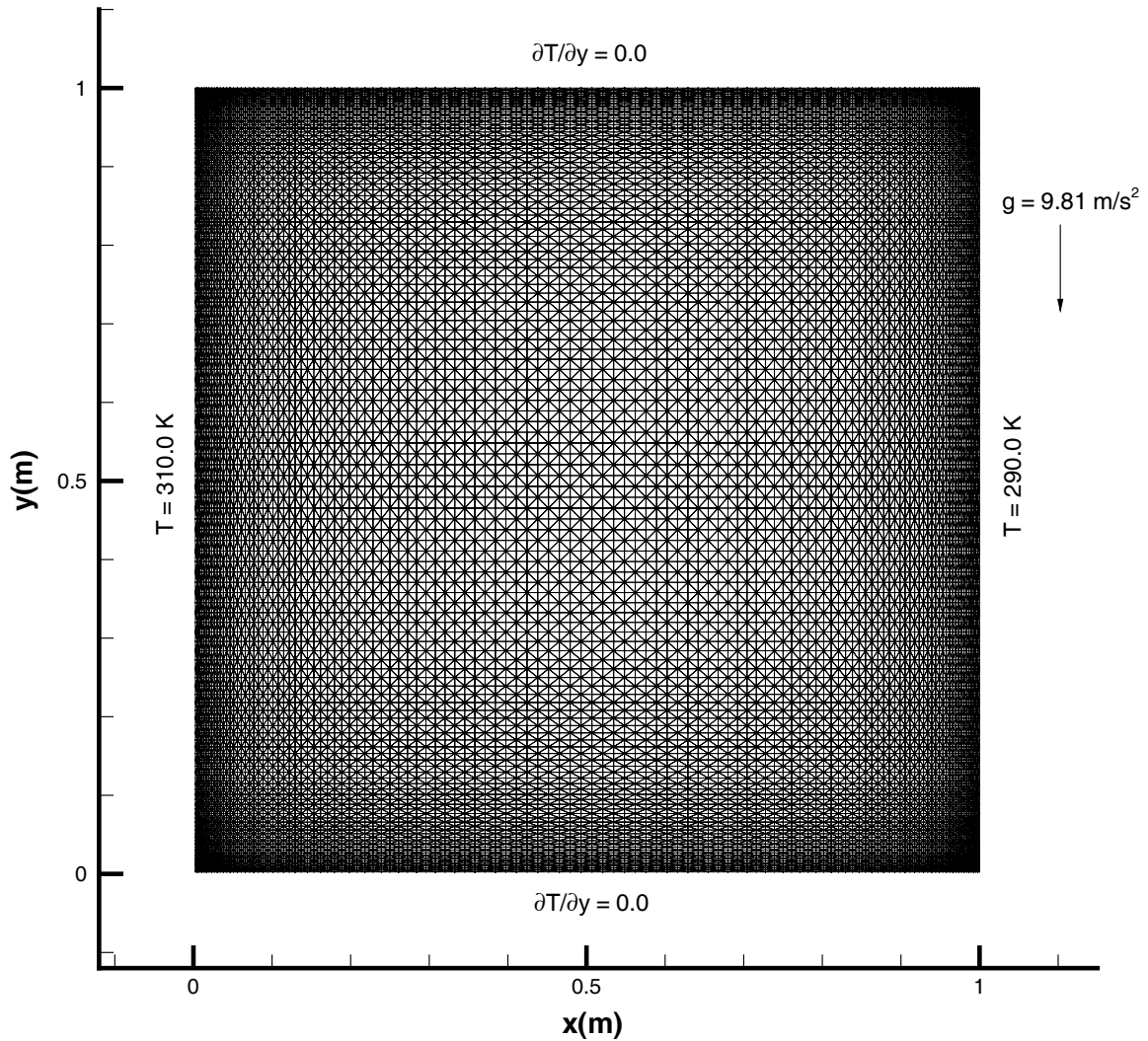


Figure 4-30 Triangular finite element mesh, geometry, and thermal boundary conditions for the thermally driven flow in a square cavity simulation.

of $Ra = 1.0 \times 10^5$, 5.0×10^5 , and 1.0×10^6 , the constant values of dynamic viscosity are $\mu = 2.553 \times 10^{-3}$ (N·s)/m², $\mu = 1.141 \times 10^{-3}$ (N·s)/m², and $\mu = 8.070 \times 10^{-4}$ (N·s)/m², respectively.

Figure 4-31 through Figure 4-34 are the PCICE-FEM solution results for pressure, temperature, density, and Mach number and streamlines, respectively, for the $Ra = 1.0 \times 10^5$ simulation. At steady-state, the pressure assumes a nearly hydrostatic profile with the simulated pressure from the top to the bottom of the domain approximately equal to 14 Pa. For the initial density and gravity constant, the analytic hydrostatic pressure drop in this domain is 11.8 Pa. Figure 4-34 is a graphical representation of the flooded Mach number contours and streamlines. The streamlines define a single vortex whose center nearly coincides with the

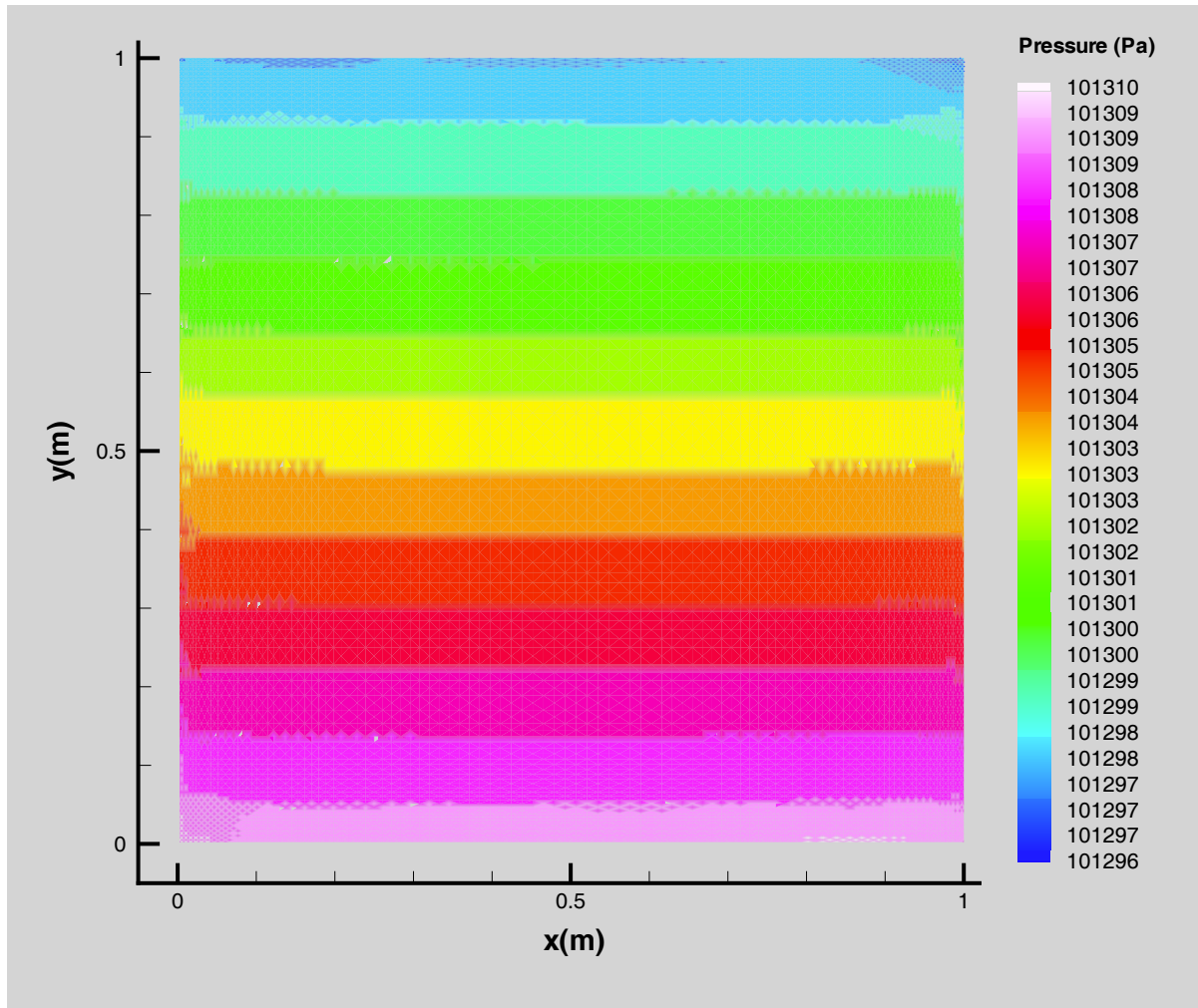


Figure 4-31 PCICICE-FEM pressure solution for the thermally driven flow in a square cavity at $Ra = 1.0 \times 10^5$.

center of the square domain. With the single vortex configuration, the temperature and density distributions shown in Figure 4-32 and Figure 4-33, respectively, are as expected. With a nearly constant pressure distribution, the temperature is a congruent image of the density. Figure 4-35 and Figure 4-36 are representations of the PCICICE-FEM temperature and Mach number/streamline solution for the $Ra = 5.0 \times 10^5$ simulation. In this case, the streamlines define three stable vortices. Figure 4-37 and Figure 4-38 are representations of the PCICE-FEM temperature and Mach number/streamline solution for the $Ra = 1.0 \times 10^6$ simulation. For this last case, the streamlines define seven stable vortices.

These three simulations of the thermally driven flow in a square cavity are computationally difficult for the PCICE-FEM scheme. Maximum fluid velocities for all three simulations is less than 0.3 m/s. Therefore, the explicit diffusion terms of the semi-implicit governing equations tend to greatly restrict the time step size of the criteria in equation (3.106). Also, the safety factor found in equation (3.106) had to be reduced to $0.05 < \beta < 0.1$ to achieve a stable

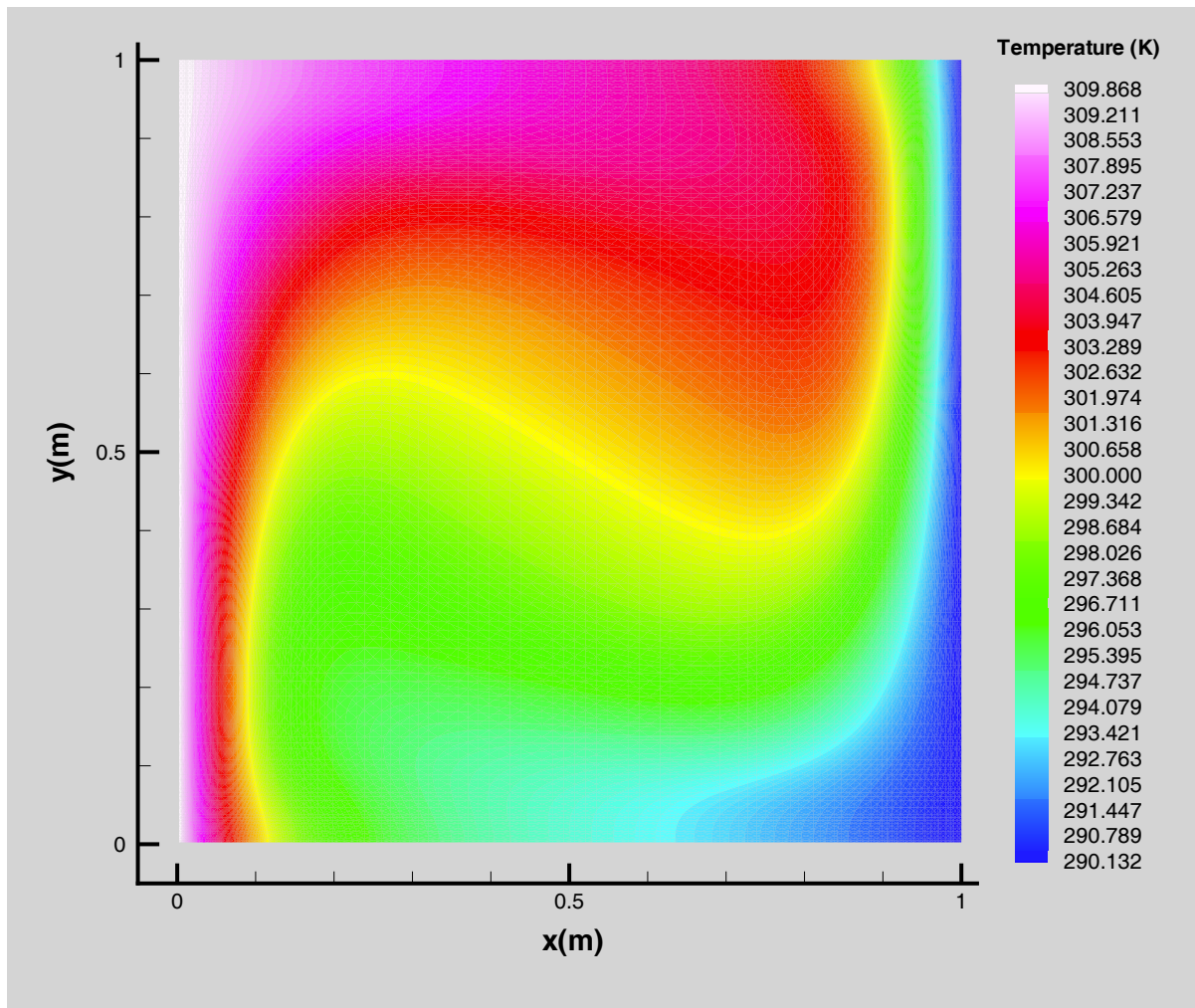


Figure 4-32 PCICICE-FEM temperature solution for the thermally driven flow in a square cavity at $Ra = 1.0 \times 10^5$.

time step. As a result, almost 100,000 time steps were required to achieve steady-state convergence. The best convergence for the simulations appeared to be with the semi-implicit time weighting factor set to $\varphi = 0.75$ and the residual smoothing coefficient set to $\varepsilon = -0.01$. With the change in pressure nearly non-existent early on in the simulation, the change in velocity was also used as a steady-state indicator of solution convergence. However, Bi-CGSTAB only required 2 iterations to converge the pressure Poisson equation as this is the minimum number of iterations coded in that specific subroutine. Other than a constraint on the equation of state, pressure seems to be irrelevant to this problem when the governing hydrodynamic equations are compressible. With change in pressure non-existent, the simulation essentially reduces from a semi-implicit simulation to an explicit simulation. This, by no means, implies that the simulation results presented here are inaccurate. It just means that this problem may be better served with a fully implicit formulation.

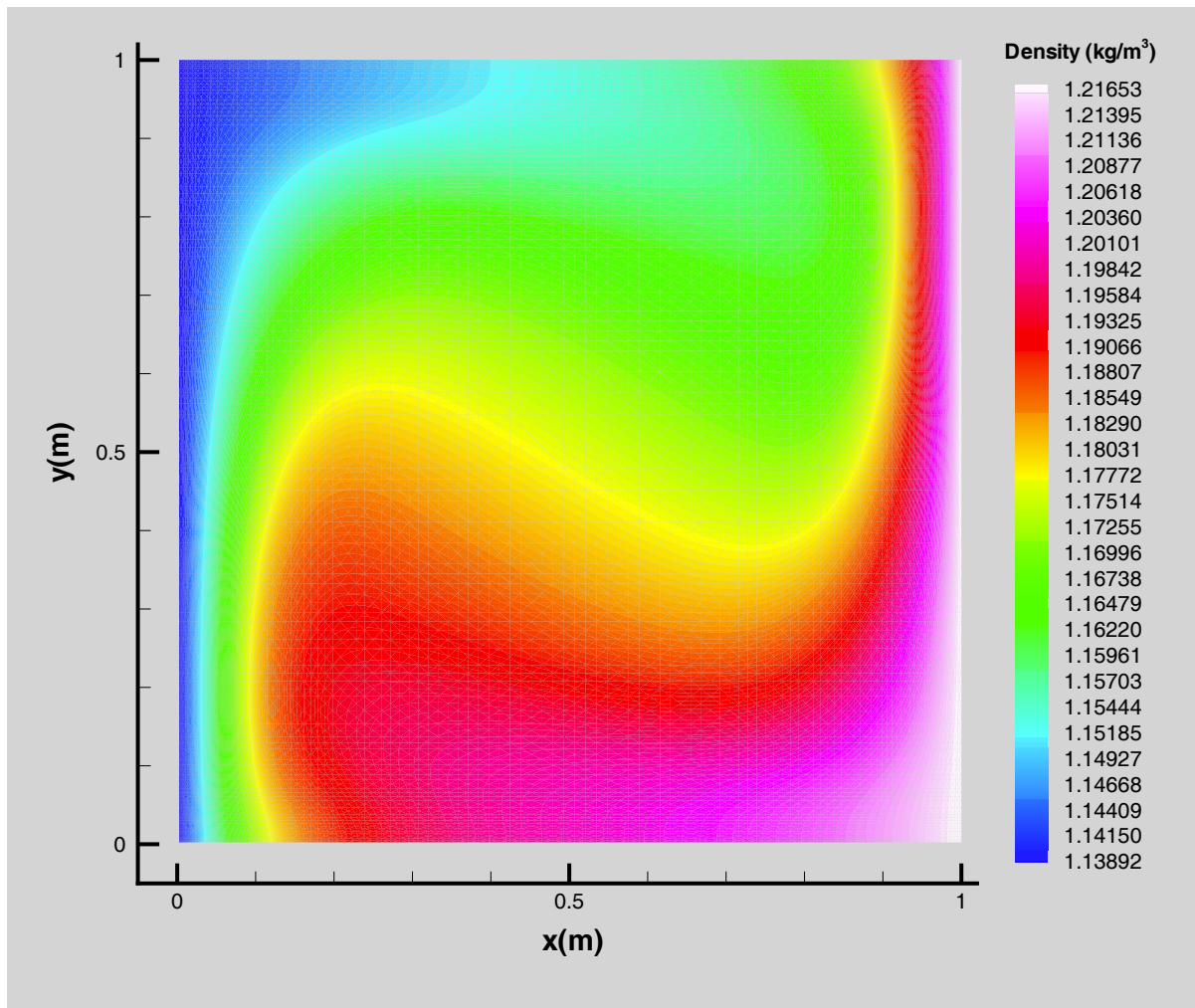


Figure 4-33 PCICE-FEM density solution for the thermally driven flow in a square cavity at $Ra = 1.0 \times 10^5$.

When the PCICE-FEM simulation results for the thermally driven flow problem presented here is compared to the incompressible Boussinesq results documented in Davis (1983), there are significant differences in the steady-state flow fields. The $Ra = 1.0 \times 10^5$ flow field presented here contains only one vortex whereas the flow field for this case found in Davis contains two vortices. The $Ra = 1.0 \times 10^5$ flow field presented here qualitatively appears more like the order of magnitude smaller $Ra = 1.0 \times 10^4$ flow field case illustrated in Davis, which also contained one vortex. For the $Ra = 1.0 \times 10^6$ case, the results presented here contained seven vortices while the flow field for this case found in Davis contained three vortices. While it might be argued that the finer mesh employed here (121×121 versus the 81×81 grid in the published results) better resolves the flow field for the $Ra = 1.0 \times 10^6$ case, it does not explain why the PCICE-FEM solution contained only one vortex for the $Ra = 1.0 \times 10^5$ case versus two for the flow field documented in Davis. It is more easily argued that the solution differences were due to the compressible versus incompressible governing hydrodynamic equations. Density in the PCICE-FEM solutions varied by approximately 8% (see Figure 4-33) across

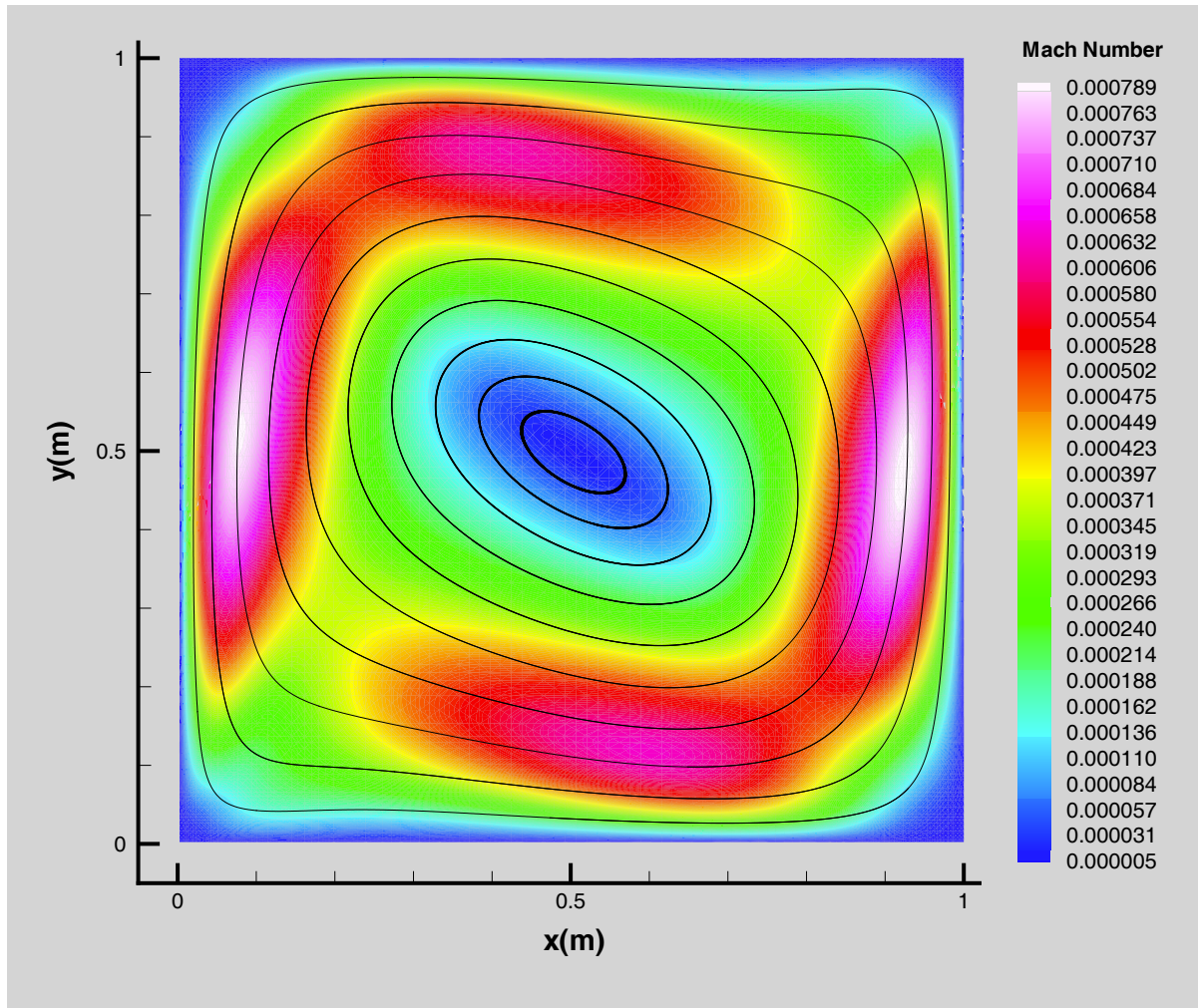


Figure 4-34 PCICE-FEM Mach number and streamline solution for the thermally driven flow in a square cavity at $Ra = 1.0 \times 10^5$.

the domain for a $\Delta T = 20.0$ K. An 8% variation in the density seems to indicate that a variable density formulation is required versus the incompressible Boussinesq approximation. It may be that a smaller domain ΔT is required for the Boussinesq approximation to be applicable.

4.3 Conclusions

A new semi-implicit pressure-based Computational Fluid Dynamics (CFD) scheme for simulating a wide range of transient and steady, inviscid and viscous compressible flow on unstructured finite elements is presented here. This new CFD scheme, termed the PCICE-FEM (Pressure-Corrected ICE-Finite Element Method) scheme, is composed of three computational phases, an explicit predictor, an elliptic pressure Poisson solution, and a semi-implicit pressure-correction of the flow variables. The PCICE-FEM scheme is capable of second-order temporal accuracy by incorporating a combination of a time-weighted form of

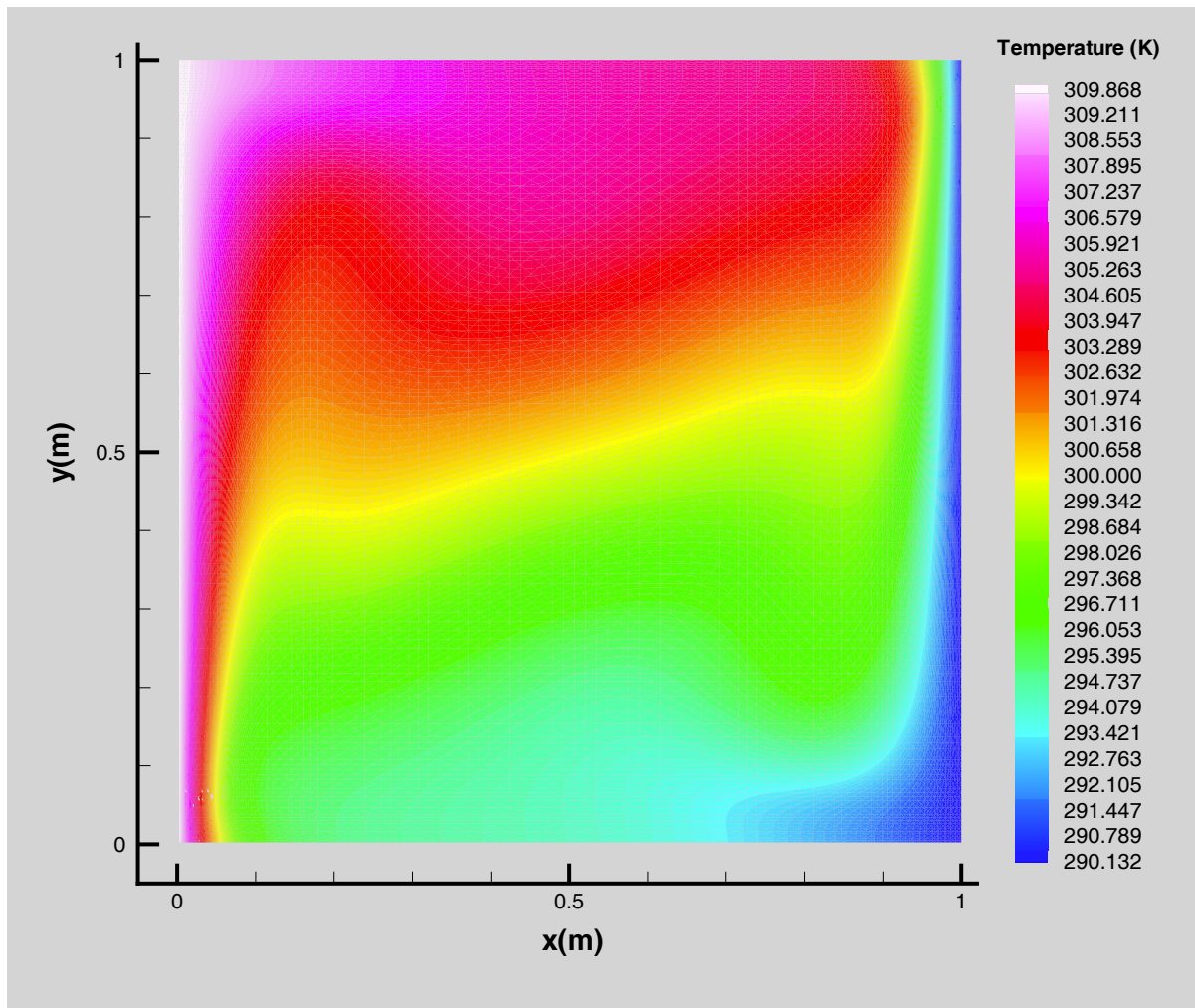


Figure 4-35 PCICICE-FEM temperature solution for the thermally driven flow in a square cavity at $Ra = 5.0 \times 10^5$.

the two-step Taylor-Galerkin Finite Element Method (FEM) scheme as an explicit predictor for the balance of momentum equations and the finite element form of a time-weighted trapezoid rule method for the semi-implicit form of the governing hydrodynamic equations. Second-order spatial accuracy is accomplished by linear unstructured finite element discretization. The PCICE-FEM scheme employs Flux-Corrected Transport (FCT) as a high-resolution filter for shock capturing. The scheme is capable of simulating flows from the nearly incompressible to the high supersonic flow regimes.

The PCICE-FEM scheme is a descendent of the Implicit Continuous-fluid Eulerian (ICE) scheme originally developed at the Los Alamos National Laboratory (LANL) by Harlow and Amsden (1971). The ICE scheme treats pressure, momentum, and density implicitly which eliminates the acoustic component of the explicit time step size criteria. The center of the ICE scheme is a pressure Poisson equation resulting from strong coupling of the conservative forms of the balance of momentum and mass conservation equations. This coupling

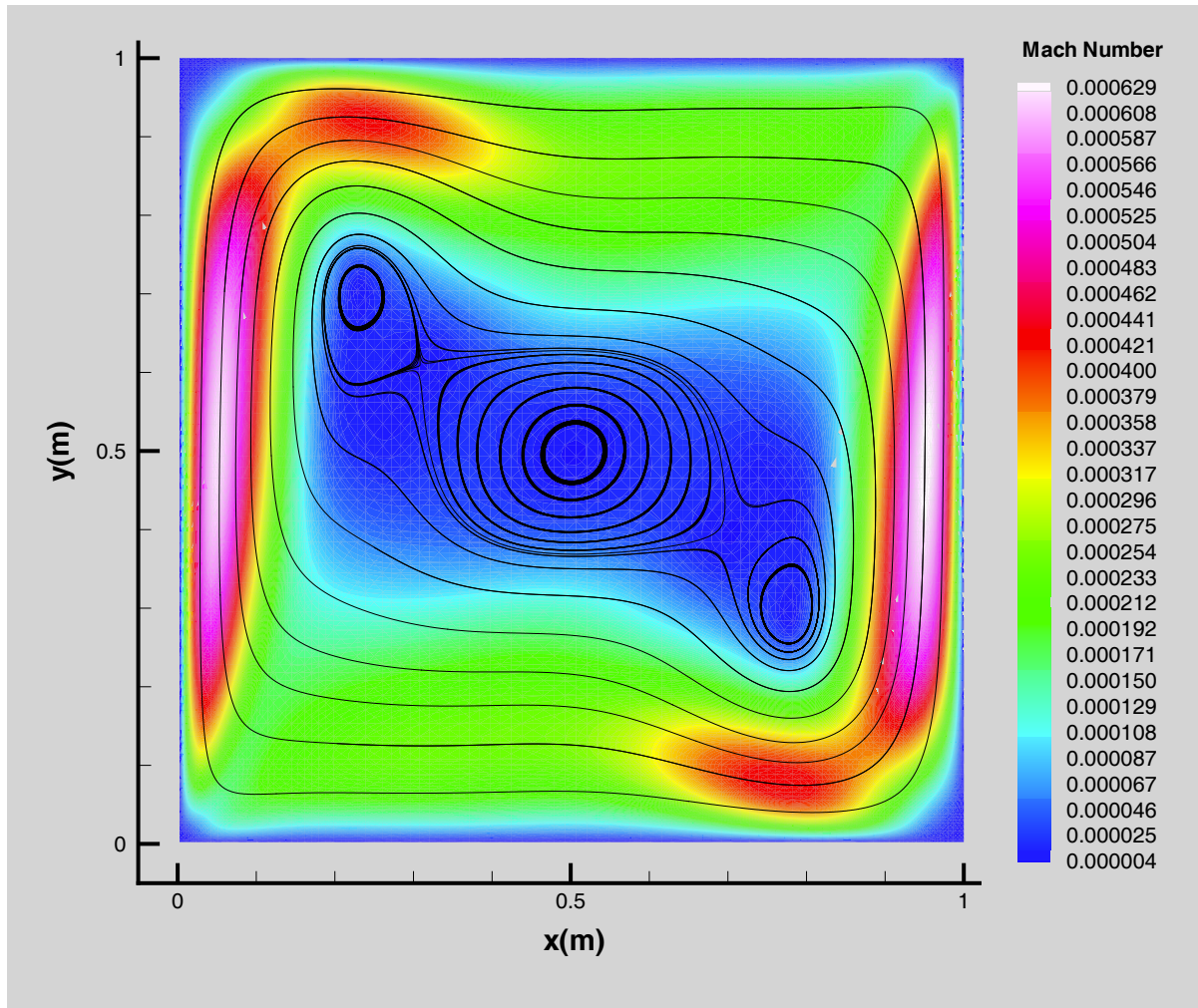


Figure 4-36 PCICICE-FEM Mach number and streamline solution for the thermally driven flow in a square cavity at $Ra = 5.0 \times 10^5$.

technique, termed here as mass-momentum coupling, is performed by substituting the balance of momentum equations directly into the mass flux term of the mass conservation equation in order to eliminate momentum as an unknown. An equation of state, expressing the change in density in terms of the change in pressure, is then used to eliminate density as an unknown. The resulting Poisson equation is in terms of pressure as the only unknown variable. Hence, the scheme is commonly referred to as pressure-based.

The solution of the pressure Poisson equation essentially solves the balance of momentum and mass conservation equations simultaneously. The pressure field results in a momentum field that satisfies mass conservation in the incompressible limit. The conservation form of the total energy equation is then solved in an explicit fashion with the implicit variables in the total energy flux term to complete the time step. This explicit treatment of total energy results in a weak coupling between the total energy and pressure fields. Various forms of the ICE scheme

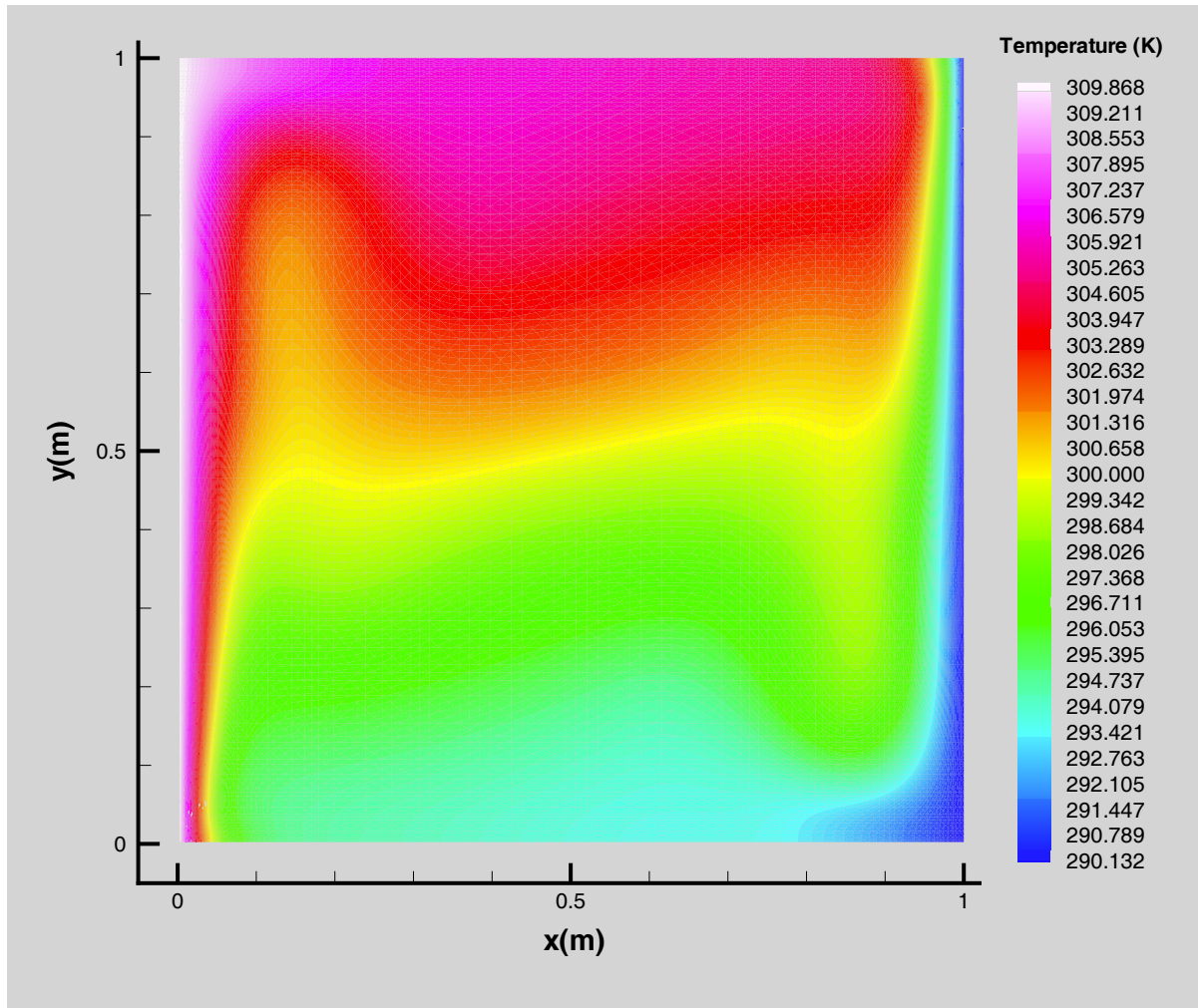


Figure 4-37 PCICICE-FEM temperature solution for the thermally driven flow in a square cavity at $Ra = 1.0 \times 10^6$.

have evolved that incorporate an iterative process between the pressure Poisson solution and the explicit energy solution to promote stronger coupling of the two fields.

The PCICE-FEM scheme represents an advancement in mass-momentum coupled, pressure-based schemes. The governing hydrodynamic equations for this scheme are the conservative form of the balance of momentum equations (Navier-Stokes), mass conservation equation, and the total energy equation. These governing equations are temporally discretized in a time-weighted, semi-implicit form. An operator splitting process is performed along explicit and implicit operators of the semi-implicit governing equations to render the PCICE-FEM scheme in the class of predictor-corrector schemes. In the original ICE scheme, only the semi-implicit balance of momentum equations were cast in predictor-corrector form. The complete set of semi-implicit governing equations in the PCICE-FEM scheme are cast in this form, an explicit predictor phase and a semi-implicit pressure-correction phase with the elliptic pressure Poisson solution coupling the predictor-corrector phases. The result of this predictor-corrector

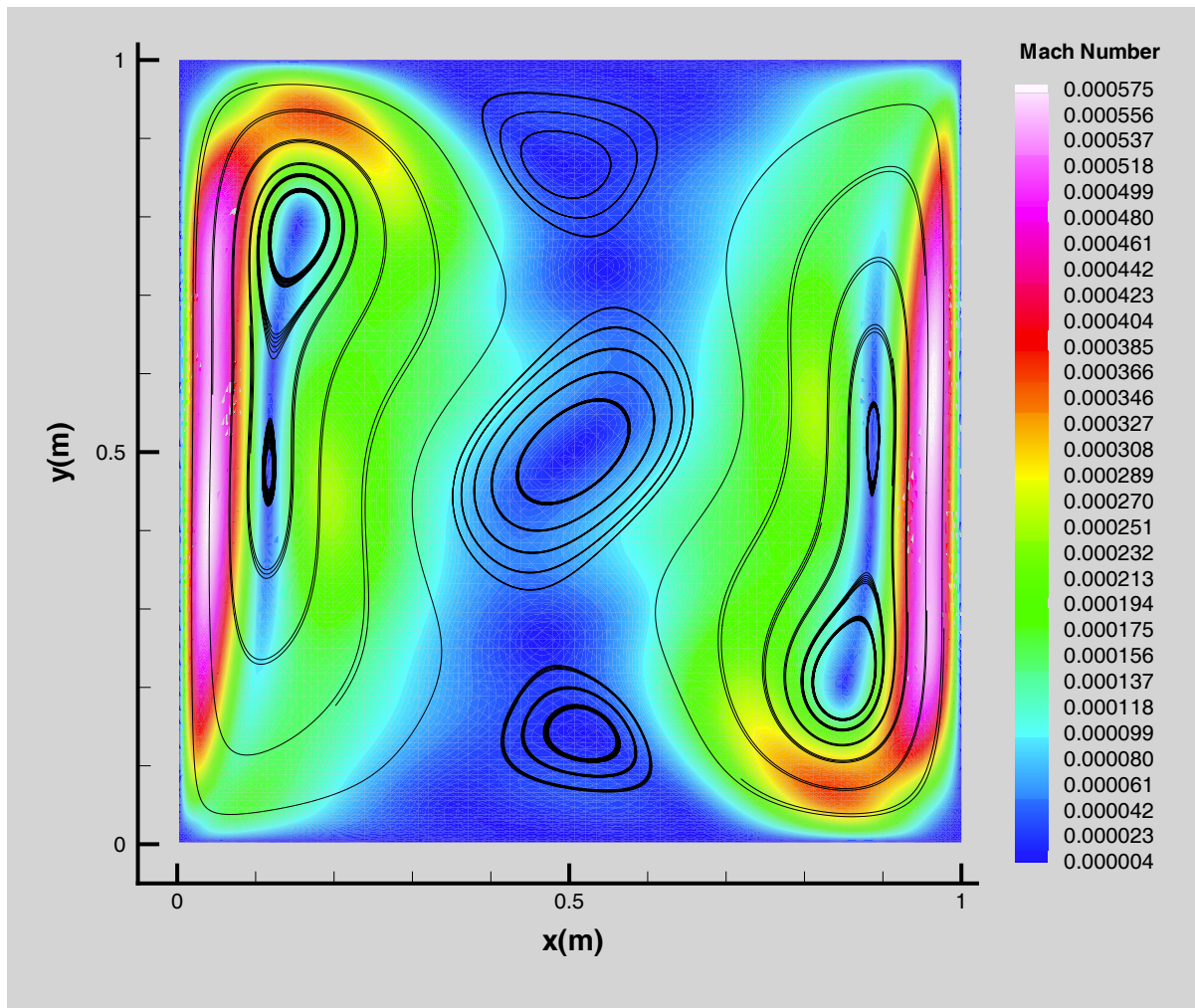


Figure 4-38 PCICE-FEM Mach number and streamline solution for the thermally driven flow in a square cavity at $Ra = 1.0 \times 10^6$.

formulation is that the pressure Poisson equation in the PCICE-FEM scheme is provided with sufficient internal energy information to avoid iteration.

The consistent-mass two-step Taylor-Galerkin FEM with FCT, the FEM-FCT scheme, is the basis for the explicit predictor phase of the PCICE-FEM scheme. A rigorous application of characteristic theory for boundary conditions has been developed here for the FEM-FCT scheme. This new application of characteristic boundary conditions is consistent with the spatial and temporal discretization of the Two-Step Taylor-Galerkin FEM scheme. What separates this characteristic boundary condition approach from previous efforts is that the same spatial order of accuracy of the governing CFD scheme is maintained over one computational cell. Therefore, the information extrapolated from multiple interior cells, which can prove to be very difficult for unstructured meshes, is avoided. This new characteristic boundary condition formulation has proven to be very robust and accurate for transient and steady-state compressible flow simulations. Simulation examples demonstrating the

robustness of the new characteristic boundary conditions for various boundary condition type are illustrated.

The ability of the PCICE-FEM scheme to accurately and efficiently simulate a wide variety of inviscid and viscous compressible flows is demonstrated here. The scheme produced the same high level of accuracy as the explicit FEM-FCT scheme when simulating strong transients (see the shock tube problem of Section 4.1.1 and the laboratory accident simulation in Section 4.1.2) and high-speed inviscid flows (see the Mach reflection problem of 4.2.1). With the acoustic component removed from the explicit time step size criteria, the PCICE-FEM scheme greatly extends (beyond the capability of the FEM-FCT scheme) the low-speed (nearly incompressible) and viscous flow regimes (see the von Karman vortex street simulation in Section 4.1.3). The PCICE-FEM scheme excels on low-speed viscous transient and steady-state simulations because of the larger time step size, residual smoothing, and the new Jacobi preconditioning matrix developed here for the Bi-CGSTAB solver.

For low-speed viscous flows, the lowest practical limit of velocity seems to be around a Mach number of $M = 0.01$. Below this limit, the time step size is severely restricted by explicit diffusion (viscous and thermal) terms (see the thermally driven flow in a square cavity problem of Section 4.2.3). For high-speed inviscid flows above a Mach number of approximately $M = 1.5$, the PCICE-FEM scheme is not as efficient as the explicit FEM-FCT scheme. The computational cost of solving the pressure Poisson equation amounts to approximately 40% of the total run time. For high Mach number flows, the denominator of the explicit time step size criteria approaches the denominator of the semi-implicit time step size criteria, $|\vec{u}| + c \rightarrow |\vec{u}|$ as the magnitude of velocity becomes much greater than the sound speed.

BIBLIOGRAPHY

A.A. Amsden and C.W. Hirt, "YAQUI: Arbitrary-Lagrangian-Eulerian Computer Program for Fluid Flows at All Speeds," Los Alamos Scientific Laboratory, LA-5100, 1973.

A.A. Amsden, P.J. O'Rourke, and T.D. Butler, "KIVA-II: A Computer Program for Chemically Reactive Flows with Sprays," Los Alamos National Laboratory, LA-11560-MS, 1989.

A.A. Amsden, "KIVA-3: A KIVA Program with Block Structured Mesh for Complex Geometries," Los Alamos National Laboratory, LA-12503-MS, 1993.

Ames Research Staff, "Equations, Tables, and Charts for Compressible Flow," NACA Report 1135, 1953.

J. Argyris, I. St. Doltsinis, and H. Friz, "Hermes Space Shuttle: Exploration of Reentry Aerodynamics", Computer Methods in Applied Mechanics and Engineering, 73, 1989.

V.S. Arpaci, Convection Heat Transfer, Prentice-Hall, Inc., 1984.

S.D. Bai, S.T. Wu, and C.W. Campbell, "Numerical Simulation of Single and Multiple Jet Interaction Within a Combustor," Proceedings, 5th International Conference in Numerical Methods in Laminar and Turbulent Flow, Montreal, 1987.

J. P. Boris and D. L. Book, "Flux-Corrected Transport. I. SHASTA, a Transport Algorithm that Works", Journal of Computational Physics, Vol. 11, 1973.

D. L. Book, J. P. Boris, and K. Hain, "Flux-Corrected Transport. II. Generalizations of the Method", Journal of Computational Physics, Vol. 18, 1975.

J. P. Boris and D. L. Book, "Flux-Corrected Transport. III. Minimal-Error FCT Algorithms," Journal of Computational Physics, Vol. 20, 1976.

M.O Bristeau, R. Glowinski, J. Periaux, and H. Viviand, "Numerical Simulation of Compressible Navier-Stokes Flows," GAMM-Workshop, Friedr. Vieweg & Sohn, 1987.

V. Casulli and D. Greenspan, "Pressure Method for the Numerical Solution of Transient, Compressible Fluid Flows," International Journal for Numerical Methods in Fluids, Vol. 4, 1984.

S. R. Chakravarthy, "Euler Equations - Implicit Schemes and Boundary Conditions," AIAA Journal, Vol. 21, 1983.

A. J. Chorin, "A Numerical Method for Solving Incompressible Viscous Problems," Journal of Computational Physics, Vol. 2, 1967.

L. D. Cloutman, C.W. Hirt, and N.C. Romero, "SOLA-ICE A Numerical Solution Algorithm for Transient Compressible Fluid Flows," Los Alamos Scientific Laboratory, LA-6236, 1976.

R. Codina, M. Vázquez, and O.C. Zienkiewicz, "A General Algorithm for Compressible and Incompressible Flow - Part III: The Semi-Implicit Form," *International Journal for Numerical Methods in Fluids*, Vol. 27, 1998.

G. de Vahl Davis, "Natural Convection of Air in a Square Cavity: A Bench Mark Numerical Solution," *International Journal for Numerical Methods in Fluids*, Vol. 3, 1983.

G. de Vahl Davis and I.P. Jones, "Natural Convection of Air in a Square Cavity: A Comparison Exercise," *International Journal for Numerical Methods in Fluids*, Vol. 3, 1983.

J. Donea, "A Taylor-Galerkin Method for Convective Transport Problems", *International Journal for Numerical Methods in Engineering*, Vol. 20, 1984.

J.K. Dukowicz, "A Particle-Fluid Numerical Model for Liquid Sprays," *Journal of Computational Physics*, Vol. 35, 1980.

M.A. Eisenberg and L.E. Malvern, "On Finite Element Integration in Natural Coordinates," *International Journal for Numerical Methods in Engineering*, Vol. 7, 1973.

M.S. Engelman, R.L. Sani, and P.M. Gresho, "The Implementation of Normal and/or Tangential Boundary Conditions in Finite Element Codes for Incompressible Fluid Flow," *International Journal for Numerical Methods in Fluids*, Vol. 2, 1982.

M.W. Evans and F.H. Harlow, "The Particle-In-Cell Method for Hydrodynamic Calculations," Los Alamos Scientific Laboratory, LA-2139, 1957.

R. Gentry, R.E. Martin, and B.J. Daly, "An Eulerian Differencing Method for Unsteady Compressible Flow Problems," *Journal of Computational Physics*, Vol. 1, 1966.

G.E. Georgiou, R. Morrow, and A.C. Metaxas, "An Improved Finite-Element Flux-Corrected Transport Algorithm," *Journal of Computational Physics*, Vol. 148, 1999.

S.K. Godunov, "Finite Difference Methods for Numerical Computation of Discontinuous Solutions of the Equations of Fluid Dynamics," *Mat. Sb.*, Vol. 47, 1959.

P.M. Gresho and R.L. Sani, "On Pressure Boundary Conditions for the Incompressible Navier-Stokes Equations," *Finite Elements in Fluids*, Vol. 7, John Wiley & Sons, 1987.

P.M. Gresho and R.L. Sani, *Incompressible Flow and the Finite Element Method*, John Wiley & Sons, 1999.

B. Gustafsson, "The Convergence Rate for Difference Approximations to Mixed Initial Boundary Value Problems," *Mathematics of Computation*, Vol. 29, 1975.

F.H. Harlow and J.E. Welch, "Numerical Calculation of Time-Dependent Viscous Incompressible Flow of Fluids with Free Surface," *Physics of Fluids*, Vol. 8, 1965a.

F.H. Harlow, J.E. Welch, J.P. Shannon, and B.J. Daly, "The MAC Method," Los Alamos Scientific Laboratory, LA-3425, 1965b.

F.H. Harlow and A.A. Amsden, "Numerical Calculation of Almost Incompressible Flow," *Journal of Computational Physics*, Vol. 3, 1968.

F.H. Harlow and A.A. Amsden, "A Numerical Fluid Dynamics Calculation for All Flow Speeds," *Journal of Computational Physics*, Vol. 8, 1971.

C. Hirsch, "Fundamentals of Numerical Discretization," Numerical Computation of Internal and External Flows, Vol. 1, John Wiley & Sons, 1988.

C. Hirsch, "Computational Methods for Inviscid and Viscous Flows," Numerical Computation of Internal and External Flows, Vol. 2, John Wiley & Sons, 1990.

C.W. Hirt, A.A. Amsden, and J.L. Cook, "An Arbitrary-Lagrangian-Eulerian Computing Method for All Flow Speeds," *Journal of Computational Physics*, Vol. 14, 1974.

C.W. Hirt, B. D. Nichols, and N.C. Romero, "SOLA - A Numerical Solution Algorithm for Transient Fluid Flows," Los Alamos Scientific Laboratory, LA-5852, 1975.

K.A Hoffman, S.T. Chiang, M.S. Siddiqui, and M. Papadakis, Fundamental Equations of Fluid Mechanics, Engineering Education SystemTM, 1996.

A. Jameson, W. Schmidt, and E. Turkel, "Numerical Solution of the Euler Equations by Finite Volume Methods using Runge-Kutta Time-Stepping Schemes," AIAA-81-1259, 1981.

A. Jameson, T.J. Baker, and N.P. Weatherhill, "Calculation of Inviscid Transonic Flow over a Complete Aircraft," AIAA-86-0103, 1986.

N.L. Johnson, "The Legacy and Future of CFD at Los Alamos," Los Alamos National Laboratory, LA-UR-96-1426, 1996.

I.P. Jones and C.P. Thomson (eds.), "Numerical Solutions for a Comparison Problem on Natural Convection in an Enclosed Cavity," AERE-R9955, HMSO, 1981.

B.A. Kashiwa, N.T. Padial, R.M. Rauenzahn, and W.B. VanderHeyden, "A Cell-Centered ICE Method of Multi-Phase Flow Simulations," in *Proceedings, ASME Symposium on Numerical Methods for Multi-Phase Flows*, Lake Tahoe, NV, 19-23 June, 1994.

A. Lapidus, "A Detached Shock Calculation by Second-Order Finite Differences", *Journal of Computational Physics*, Vol. 2, 1967.

P. D. Lax, "Weak Solutions of Non-Linear Hyperbolic Equations and Their Numerical Computation," *Communications in Pure and Applied Mathematics*, Vol. 7, 1954.

R. Löhner, K. Morgan, O.C. Zienkiewicz, "The Solution of Nonlinear Hyperbolic Equation Systems by the Finite Element Method", *International Journal for Numerical Methods in Fluids*, Vol. 4, 1984.

R. Löhner, K. Morgan, and O.C. Zienkiewicz, "An Adaptive Finite Element Procedure for Compressible High Speed Flows", *Computer Methods in Applied Mechanics and Engineering*, Vol. 51, 1985a.

R. Löhner, K. Morgan, and J. Peraire, "A Simple Extension to Multidimensional Problems of the Artificial Viscosity Due to Lapidus," *Communications in Applied Numerical Methods*, Vol. 1, 1985b.

R. Löhner, K. Morgan, J. Peraire, and M. Vahdati, "Finite Element Flux-Corrected Transport (FEM-FCT) for the Euler and Navier-Stokes Equations", *International Journal for Numerical Methods in Fluids*, Vol. 7, 1987.

R. Löhner, K. Morgan, M. Vahdati, J.P. Boris, and D.L. Book, "FEM-FCT: Combining Unstructured Grids with High Resolution", *Communications in Applied Numerical Methods*, Vol. 4, 1988.

R. Löhner, personal communication, 1992.

R. Löhner, personal communication, 1994.

R. Löhner, "Applied CFD Techniques," John Wiley & Sons, 2001.

MATHEMATICA[®], Version 3.0, Wolfram Research, Inc., 1996.

R.W. MacCormack and B.S. Baldwin, "A Numerical Method for Solving the Navier-Stokes Equations with Application to Shock-Boundary Layer Interaction," *AIAA 75-1*, 1975.

K. Morgan and J. Peraire, "Finite Element Methods for Compressible Flows," von Karman Institute for Fluid Dynamics, Lecture Series 1987-04, March 2-6, 1987.

E.S. Oran and J.P. Boris, Numerical Simulation of Reactive Flow, Elsevier Science Publishing, Co., Inc., 1987.

S.V. Patankar and D.B. Spalding, "A Calculation Procedure for Heat, Mass, and Momentum Transfer in Three-Dimensional Parabolic Flows," *International Journal of Heat and Mass Transfer*, Vol. 15, 1972.

S.V. Patankar, Numerical Heat Transfer and Fluid Flow, Hemisphere, 1980.

- G. Patnaik, R.H. Guirguis, J.P. Boris, and E.S. Oran, "A Barely Implicit Correction for Flux-Corrected Transport," *Journal of Computational Physics*, Vol. 71, 1987.
- J. Peraire, J. Peiro, L. Formaggia, K. Morgan, and O.C. Zienkiewicz, "Finite Element Euler Computations in Three Dimensions," *International Journal for Numerical Methods in Engineering*, Vol. 26, 1988.
- J.D. Ramshaw and J.K. Dukowicz, "APACHE: A Generalized-Mesh Eulerian Computer Code for Multicomponent Chemically Reactive Fluid Flow," Los Alamos National Laboratory, LA-7427, 1979.
- W.C. Rivard, O.A. Farmer, and T.D. Butler, "RICE: A Computer Program for Multi-Component Chemically Reactive Flow at All Speeds," Los Alamos Scientific Laboratory, LA-5812, 1975.
- M.A. Saad, Compressible Fluid Flow, Prentice-Hall, 1985.
- L.J. Segerlind, "Applied Finite Element Analysis," Second Edition, John Wiley & Sons, 1984.
- W.T. Sha, "COMMIX-1B: A Three Dimensional Transient Single-Phase Computer Program for Thermal Hydraulic Analysis of Single and Multi-Component Systems," Argonne National Laboratory, ANL-85-42, Vol. 1, 1985.
- R.C. Swanson and E. Turkel, "On Central-Difference and Upwind Schemes," *Journal of Computational Physics*, Vol. 101, 1992.
- TECPLOT[®], Version 8.0, Amtec Engineering, Inc., 1999.
- H. A. Van Der Vorst, "BI-CGSTAB: A Fast and Smoothly Converging Variant of BI-CG for the Solution of Non-Symmetric Linear Systems," *SIAM Journal of Scientific and Statistical Computing*, Vol. 13, No. 2, 1992.
- H. A. Van Der Vorst, personal communication, 1999.
- J.P. Van Doormaal and G.D. Raithby, "Enhancements of the SIMPLE Method for Predicting Incompressible Fluid Flows," *Numerical Heat Transfer*, Vol. 7, 1984.
- J.P. Van Doormaal and G.D. Raithby, "An Evaluation of the Segregated Approach for Predicting Incompressible Fluid Flows," *National Heat Transfer Conference*, Denver, Colorado, 1985.
- J.P. Van Doormaal, G.D. Raithby, and B.H. McDonald, "The Segregated Approach to Predicting Viscous Compressible Fluid Flows," *Journal of Turbomachinery*, Vol. 109, 1987.
- A.E.P. Veldman, "Missing Boundary Conditions? Discretize First, Substitute Next, and Combine Later," *SIAM Journal of Scientific and Statistical Computing*, Vol. 11, No. 1, 1990.

C.K. Westbrook, "A Generalized ICE Method for Chemically Reactive Flows in Combustion Systems," *Journal of Computational Physics*, Vol. 28, 1978.

F.M. White, *Viscous Fluid Flow*, 2nd ed., McGraw-Hill, Inc., 1991.

F. Xiao and T. Yabe, "A Method to Trace the Sharp Interface of Two Fluids in Calculations Involving Shocks," *Shock Waves*, Vol. 4, 1994.

F. Xiao, T. Yabe, T. Ito, and M. Tajima, "An Algorithm for Simulating Solid Objects Suspended in Stratified Flow," *Computer Physics Communications*, Vol. 102, 1997.

T. Yabe and P.Y. Wang, "Unified Numerical Procedure for Compressible and Incompressible Fluid," *Journal of the Physical Society of Japan*, Vol. 60, 1991.

T. Yabe, F. Xiao, and H. Mochizuki, "Simulation Technique for Dynamic Evaporation Process," *Nuclear Engineering and Design*, Vol. 155, 1995.

T. Yabe, F. Xiao, and Y. Zhang, "Strategy for Unified Solution of Solid, Liquid, Gas, and Plasmas, AIAA 99-3509, 1999.

S.Y. Yoon and T. Yabe, "The Unified Simulation for Incompressible and Incompressible Flow by the Predictor-Corrector Scheme based on the CIP Method," *Computer Physics Communications*, Vol. 119, 1999.

S.T. Zalesak, "Fully Multidimensional Flux-Corrected Transport Algorithms for Fluids", *Journal of Computational Physics*, Vol. 31, 1979.

O.C. Zienkiewicz and J. Wu, "Incompressibility Without Tears - How to Avoid Restrictions of Mixed Formulation," *International Journal for Numerical Methods in Engineering*, Vol. 32, 1991.

O.C. Zienkiewicz and J. Wu, "A General Explicit or Semi-Explicit Algorithm for Compressible and Incompressible Flows," *International Journal for Numerical Methods in Engineering*, Vol. 35, 1992.

O.C. Zienkiewicz and R. Codina, "A General Algorithm for Compressible and Incompressible Flow - Part I: The Split, Characteristic-Based Scheme," *International Journal for Numerical Methods in Fluids*, Vol. 20, 1995.

O.C. Zienkiewicz, K. Morgan, B.V.K. Satya Sai, R. Codina, and M. Vázquez, "A General Algorithm for Compressible and Incompressible Flow - Part II: Tests on the Explicit Form," *International Journal for Numerical Methods in Fluids*, Vol. 20, 1995.

O.C. Zienkiewicz, P. Nithiarasu, R. Codina, M. Vázquez, and P. Ortiz, "The Characteristic-Based-Split Procedure: An Efficient and Accurate Algorithm for Fluid Problems," *International Journal for Numerical Methods in Fluids*, Vol. 31, 1999.

APPENDIX A: General Properties of Linear Triangular Finite Elements

A.1 Properties of Shape Functions for Linear Triangular Finite Elements

The linear triangular element shown in Figure 6-1 has straight sides and three nodes, one at each corner. A consistent labeling of the nodes is a necessity and proceeds in a counter-

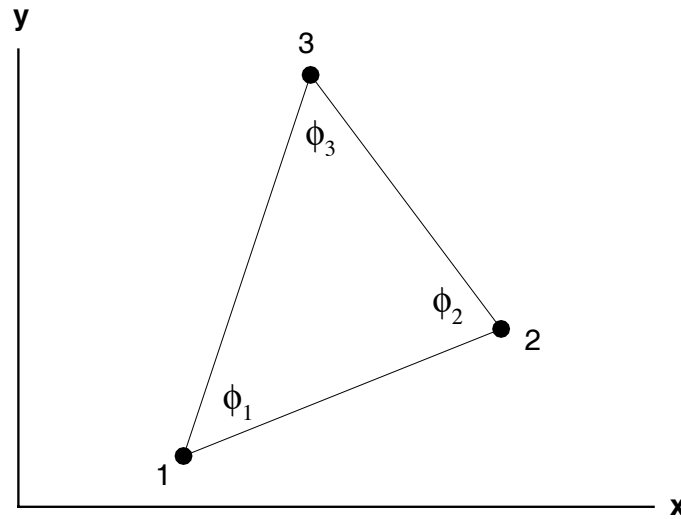


Figure 6-1 Parameters for the linear triangular element.

clockwise fashion in accordance with the right-hand rule of vectors. The labeling of node 1 is arbitrary. The nodal values of ϕ are ϕ_1 , ϕ_2 , and ϕ_3 whereas the nodal coordinates are (x_1, y_1) , (x_2, y_2) , and (x_3, y_3) . The variable ϕ is independent of the spatial coordinates of the nodes. The following discussion of finite element shape functions for linear triangles is paraphrased from Segerlind (1984).

The variation of ϕ in the domain can be defined by the interpolation polynomial

$$\phi = \alpha_1 + \alpha_2 x + \alpha_3 y \quad (0.1)$$

with the nodal conditions

$$\phi = \phi_1 \text{ at } x = x_1, y = y_1$$

$$\phi = \phi_2 \text{ at } x = x_2, y = y_2$$

$$\phi = \phi_3 \text{ at } x = x_3, y = y_3.$$

Substitution of these conditions into equation (0.1) produces the system of equations

$$\begin{aligned}\phi_1 &= \alpha_1 + \alpha_2 x_1 + \alpha_3 y_1 \\ \phi_2 &= \alpha_1 + \alpha_2 x_2 + \alpha_3 y_2 \\ \phi_3 &= \alpha_1 + \alpha_2 x_3 + \alpha_3 y_3\end{aligned}\tag{0.2}$$

which can be recast as

$$\begin{Bmatrix} \phi_1 \\ \phi_2 \\ \phi_3 \end{Bmatrix} = \begin{bmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{bmatrix} \begin{Bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \end{Bmatrix}.\tag{0.3}$$

The determinant of the coefficient matrix in equation (0.3) is

$$\det \begin{bmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{bmatrix} = 2A$$

where A is the area of the triangle. Solving the system of equations (0.3) for ϕ_1 , ϕ_2 , and ϕ_3 yields

$$\begin{aligned}\alpha_1 &= \frac{1}{2A} [(x_2 y_3 - x_3 y_2) \phi_1 + (x_3 y_1 - x_1 y_3) \phi_2 + (x_1 y_2 - x_2 y_1) \phi_3] \\ \alpha_2 &= \frac{1}{2A} [(y_2 - y_3) \phi_1 + (y_3 - y_1) \phi_2 + (y_1 - y_2) \phi_3] \\ \alpha_3 &= \frac{1}{2A} [(x_3 - x_2) \phi_1 + (x_1 - x_3) \phi_2 + (x_2 - x_1) \phi_3].\end{aligned}\tag{0.4}$$

Substituting equation (0.4) into equation (0.1) and rearranging produces an equation for ϕ in terms of three shape functions, N_1 , N_2 , and N_3 , and the nodal values of ϕ , ϕ_1 , ϕ_2 , and ϕ_3 ,

$$\phi = N_1 \phi_1 + N_2 \phi_2 + N_3 \phi_3\tag{0.5}$$

where

$$N_1 = \frac{1}{2A} [a_1 + b_1 x + c_1 y] \quad (0.6)$$

$$N_2 = \frac{1}{2A} [a_2 + b_2 x + c_2 y] \quad (0.7)$$

$$N_3 = \frac{1}{2A} [a_3 + b_3 x + c_3 y] \quad (0.8)$$

and

$$a_1 = x_2 y_3 - x_3 y_2, \quad b_1 = y_2 - y_3, \quad \text{and} \quad c_1 = x_3 - x_2$$

$$a_2 = x_3 y_1 - x_1 y_3, \quad b_2 = y_3 - y_1, \quad \text{and} \quad c_2 = x_1 - x_3$$

$$a_3 = x_1 y_2 - x_2 y_1, \quad b_3 = y_1 - y_2, \quad \text{and} \quad c_3 = x_2 - x_1.$$

Thus, the independent variable ϕ is related to the nodal values by a set of shape functions that are linear in x and y . This means that the derivatives of ϕ are constant within an element. For example, differentiating equation (0.5) with respect x gives

$$\frac{\partial \phi}{\partial x} = \frac{\partial N_1}{\partial x} \phi_1 + \frac{\partial N_2}{\partial x} \phi_2 + \frac{\partial N_3}{\partial x} \phi_3 \quad (0.9)$$

with

$$\frac{\partial N_i}{\partial x} = \frac{b_i}{2A} \quad i = 1, 2, \text{ and } 3.$$

Therefore,

$$\frac{\partial \phi}{\partial x} = \frac{1}{2A} [b_1 \phi_1 + b_2 \phi_2 + b_3 \phi_3]. \quad (0.10)$$

Because b_1 , b_2 , and b_3 are elemental constants and ϕ_1 , ϕ_2 , and ϕ_3 are independent of the space coordinates, the derivative has a constant value in a linear triangular finite element.

A natural coordinate system for linear triangular finite elements is area coordinates, which are denoted by L_1 , L_2 , and L_3 . These coordinates give the ratio of the area of a sub triangular

region to the area of the complete triangle. Consider point B as shown in Figure 6-2. The first

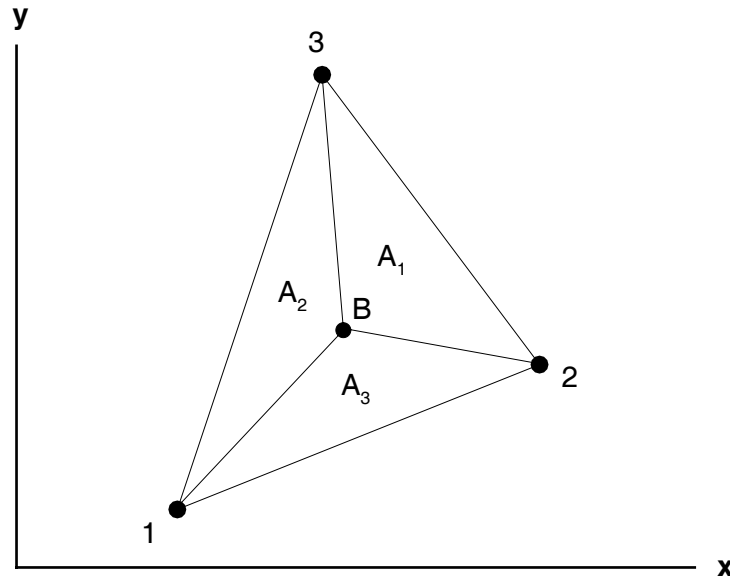


Figure 6-2 A triangular element divided into areas corresponding to the area coordinates.

area coordinate is found by forming the ratio

$$\frac{A_1}{A} = L_1 \quad (0.11)$$

which is the ratio of the area of the sub triangular region, denoted by A_1 in Figure 6-2, by the total area A of the element. Similar equations can be written for L_2 and L_3 giving

$$L_2 = \frac{A_2}{A} \text{ and } L_3 = \frac{A_3}{A}. \quad (0.12)$$

Because $A_1 + A_2 + A_3 = A$,

$$L_1 + L_2 + L_3 = 1. \quad (0.13)$$

Equation (0.11) can be recast into another form. Multiplying the numerator and denominator by 2 gives

$$\frac{2A_1}{2A} = L_1. \quad (0.14)$$

Using the determinant for the region corresponding to L_1 produces

$$\det \begin{bmatrix} 1 & x & y \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{bmatrix} = 2A_1 \quad (0.15)$$

or

$$2A_1 = (x_2y_3 - x_3y_2) + (y_2 - y_3)x + (x_3 - x_2)y \quad (0.16)$$

where x and y are the coordinates of B in Figure 6-2. Substituting equation (0.16) into equation (0.14) yields

$$L_1 = \frac{1}{2A} [(x_2y_3 - x_3y_2) + (y_2 - y_3)x + (x_3 - x_2)y]. \quad (0.17)$$

Equation (0.17) is identical to equation (0.6). Thus,

$$N_1 = L_1$$

and similarly for L_2 and L_3 shows that

$$N_2 = L_2 \text{ and } N_3 = L_3.$$

Therefore, the area coordinates for the linear triangular element are identical to the shape functions and the two sets of quantities can be interchanged.

The advantage of using the area coordinate system is the existence of an integration formula that simplifies the evaluation of area integrals (Eisenberg and Malvern, 1973). This integral equation is

$$\int_A L_1^i L_2^j L_3^k dA = 2A \frac{(i)!(j)!(k)!}{(i+j+k+2)!}, \quad (0.18)$$

where i, j , and k are exponential powers.

A.2 Evaluation of Basic Integrals Common to Linear Triangular Finite Elements

Equation (0.18) allows the various finite element integrals found in Chapters 2 and 3 to be exactly integrated for linear triangles. Therefore, the time consuming (and less accurate) numerical integration (Gauss quadrature) can be avoided. The following finite element volume integrals are evaluated on an elemental basis with equation (0.18):

$$\int_{\Omega} d\Omega = 2A \frac{(0)!}{(2)!} = A \quad (0.19)$$

$$\int_{\Omega} \{N\} d\Omega = 2A \frac{1}{(3)!} \begin{Bmatrix} (1)! \\ (1)! \\ (1)! \end{Bmatrix} = \frac{A}{3} \begin{Bmatrix} 1 \\ 1 \\ 1 \end{Bmatrix}$$

$$\int_{\Omega} \{N\}[N] d\Omega = 2A \frac{1}{(4)!} \begin{bmatrix} (2)! & (1)! & (1)! \\ (1)! & (2)! & (1)! \\ (1)! & (1)! & (2)! \end{bmatrix} = \frac{A}{12} \begin{bmatrix} 2 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 2 \end{bmatrix} \quad (0.20)$$

$$\int_{\Omega} \{N\}[\vec{\nabla} N] d\Omega = \int_{\Omega} \{N\} d\Omega [\vec{\nabla} N] = 2A \frac{1}{(3)!} \begin{Bmatrix} (1)! \\ (1)! \\ (1)! \end{Bmatrix} [\vec{\nabla} N] \quad (0.21)$$

$$= \frac{A}{3} \begin{bmatrix} \vec{\nabla} N_1 & \vec{\nabla} N_2 & \vec{\nabla} N_3 \\ \vec{\nabla} N_1 & \vec{\nabla} N_2 & \vec{\nabla} N_3 \\ \vec{\nabla} N_1 & \vec{\nabla} N_2 & \vec{\nabla} N_3 \end{bmatrix}$$

$$\begin{aligned}
\int_{\Omega} \{\vec{\nabla} N\} \cdot [\vec{\nabla} N] d\Omega &= \int_{\Omega} d\Omega \{\vec{\nabla} N\} \cdot [\vec{\nabla} N] = 2A \frac{(0)!}{(2)!} \{\vec{\nabla} N\} \cdot [\vec{\nabla} N] \\
&= A \begin{bmatrix} \vec{\nabla} N_1 \cdot \vec{\nabla} N_1 & \vec{\nabla} N_1 \cdot \vec{\nabla} N_2 & \vec{\nabla} N_1 \cdot \vec{\nabla} N_3 \\ \vec{\nabla} N_2 \cdot \vec{\nabla} N_1 & \vec{\nabla} N_2 \cdot \vec{\nabla} N_2 & \vec{\nabla} N_2 \cdot \vec{\nabla} N_3 \\ \vec{\nabla} N_3 \cdot \vec{\nabla} N_1 & \vec{\nabla} N_3 \cdot \vec{\nabla} N_2 & \vec{\nabla} N_3 \cdot \vec{\nabla} N_3 \end{bmatrix}
\end{aligned} \tag{0.22}$$

Note that shape function derivative matrices may be brought outside of integrals as they are defined as elemental constants for linear triangular finite elements. The following two boundary integrals are evaluated as line integrals.

$$\int_{\Gamma} \{N\} d\Gamma = \frac{l_b}{2} \begin{Bmatrix} 1 \\ 1 \end{Bmatrix} \tag{0.23}$$

$$\int_{\Gamma} \{N\} [N] d\Gamma = \frac{l_b}{6} \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \tag{0.24}$$

In equations (0.23) and (0.24), l_b is the length of the face of the element on the boundary.

A.3 Boundary Face Outward Pointing Unit Normal Vector and Element Length

Consider Figure 6-3. The boundary face defined by local nodes 1 and 2 of element E has a outward pointing unit normal vector given by

$$\vec{\eta} = \frac{-\vec{\nabla} N_3}{|\vec{\nabla} N_3|} \tag{0.25}$$

where N_3 is the shape function adjacent to nodes 1 and 2. Note that the local node numbering of element E is arbitrary. Thus, the boundary face outward pointing unit normal vector is evaluated by the shape function derivative adjacent to the boundary face.

The element length h_3 from node 3 to the opposite boundary face connecting nodes 1 and 2 in a perpendicular direction is given by

$$h_3 = \frac{1}{|\vec{\nabla} N_3|}. \tag{0.26}$$

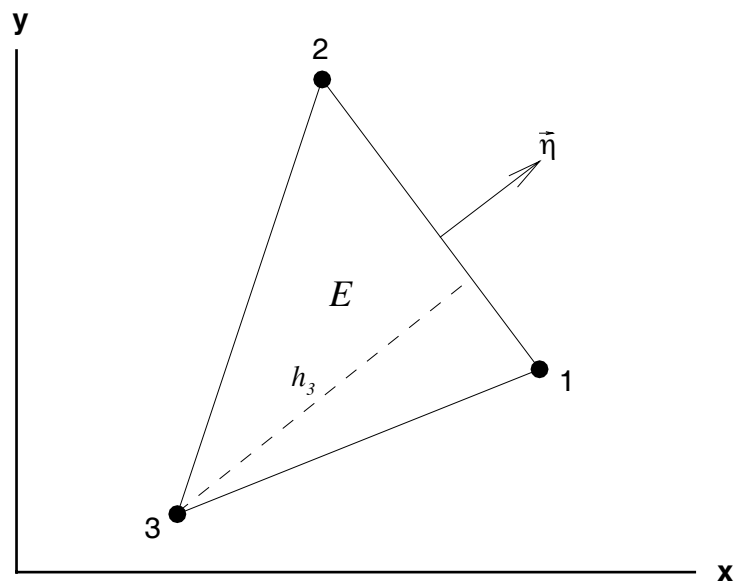


Figure 6-3 Boundary face unit outward normal and element length parameters.

Similar lengths for nodes 1 and 2 can be determined. The average of these three lengths gives the average element length used in determine time step stability criteria and diffusion coefficients.

APPENDIX B: Determination of Boundary Node Outward Pointing Unit Normal Vector that is Consistent with the Principle of Mass Conservation

There are various techniques for implementing normal and/or tangential boundary conditions for the FEM. For the mass-momentum coupling procedure employed for the PCICE-FEM scheme, the resulting pressure Poisson equation is essentially a mass conservation equation. Therefore, it follows that the principle of global mass conservation should be used to define a unique direction for the outward pointing unit normal vector at any node on the boundary of a discretized domain. This principle permits the consistent application of essential or natural boundary conditions on the discretized boundary with out regard to boundary shape or orientation (invariant) with respect to the coordinate directions. For the PCICE-FEM scheme, it is crucial that the explicit components and the semi-implicit pressure-correction components for the balance of momentum equations adhere to the principle of mass conservation along free-slip solid walls and symmetry planes. The divergence of the explicit balance of momentum components constitute a major portion of the right-hand side of the pressure Poisson equation. The implicit pressure obtained from the solution of the pressure Poisson equation is the sole correction variable for the semi-implicit pressure-correction equations for the balance of momentum. The following formulation was presented by Engelman et al. (1982) for the determination of nodal outward pointing unit normal vectors for discretized finite element boundaries. It is applicable for any multi-dimensional finite element formulation regardless of element type, element shape, or order of interpolation.

Consider the problem of a boundary along which either the normal or tangential balance of momentum (or velocity) component is specified while the remaining component is free, as depicted in Figure 6-4. Working with the components of velocity as, u and v , as the degrees of freedom for node i on the boundary, the local x - y coordinate system at node i is rotated to coincide with the normal and tangential directions, $\vec{\eta}$ and $\vec{\tau}$, respectively. The normal velocity u_η and the tangential velocity u_τ now become the degrees of freedom at node i . The normal velocity u_η and the tangential velocity u_τ are defined by

$$\begin{aligned}\vec{u} \cdot \vec{\eta} &= u_\eta = \eta_x u + \eta_y v \\ \vec{u} \cdot \vec{\tau} &= u_\tau = \tau_x u + \tau_y v,\end{aligned}\tag{0.27}$$

where η_x and η_y are the x and y components of $\vec{\eta}$, respectively, and τ_x and τ_y are the x and y components of $\vec{\tau}$, respectively. Also, the unit tangential vector can be put in terms of the unit normal vector components. Referring to Figure 6-4 and maintaining the right-hand rule of vectors, $\tau_x = -\eta_y$ and $\tau_y = \eta_x$ such that the tangential velocity written in terms of the unit normal vector components is

$$u_\tau = -\eta_y u + \eta_x v.\tag{0.28}$$

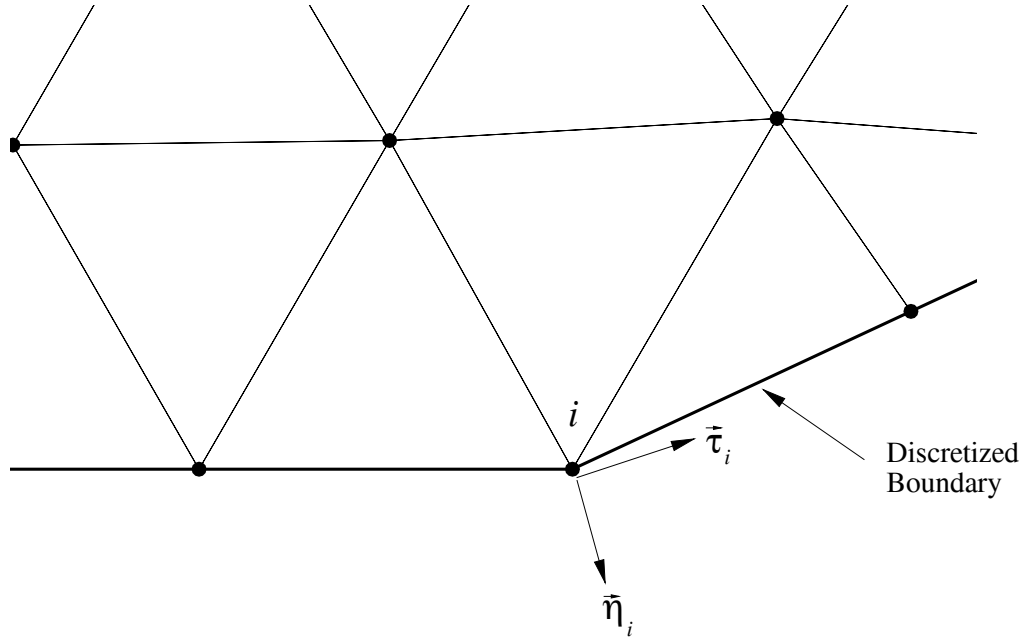


Figure 6-4 Normal and tangential directions on a finite element boundary node.

The conversion from (u, v) to (u_η, u_τ) is achieved at node i by the orthogonal rotation matrix R_i defined by

$$\begin{Bmatrix} u_\eta \\ u_\tau \end{Bmatrix}_i = R_i \begin{Bmatrix} u \\ v \end{Bmatrix}_i, \quad (0.29)$$

where

$$R_i = \begin{bmatrix} \eta_x & \eta_y \\ -\eta_y & \eta_x \end{bmatrix}_i. \quad (0.30)$$

Being an orthogonal matrix, R_i has the property of $R_i^{-1} = R_i^T$,

$$R_i^T = \begin{bmatrix} \eta_x & -\eta_y \\ \eta_y & \eta_x \end{bmatrix}_i, \quad (0.31)$$

so that the conversion from (u_η, u_τ) to (u, v) is

$$\begin{Bmatrix} u \\ v \end{Bmatrix}_i = R_i^T \begin{Bmatrix} u_\eta \\ u_\tau \end{Bmatrix}_i. \quad (0.32)$$

Along a curved boundary discretized by linear elements (see Figure 6-4), there is a discontinuity in the slope of the discretized boundary from element to element. The normal direction at a node i is then not uniquely defined (non-uniqueness can also occur on the exact boundary, such as at the corner of a step). It could vary between the boundary face normals of the adjacent elements to node i without some form of an appropriate constraint. A unique and appropriate normal direction may be determined by invoking mass conservation arguments when dealing with an incompressible fluid. Using the weighted residual form of the mass conservation equation for an incompressible fluid is

$$\int_{\Omega} \psi_i \vec{\nabla} \cdot \vec{u} d\Omega = 0 \quad (0.33)$$

for all ψ_i , where ψ_i are the basis functions for pressure and Ω is the entire discretized domain. Because the space of pressure basis functions must satisfy a constant pressure field, it follows from equation (0.33) that a global mass balance is always satisfied, i.e.

$$\int_{\Omega} \vec{\nabla} \cdot \vec{u} d\Omega = 0. \quad (0.34)$$

Inserting the FEM approximate solution,

$$\vec{u} \approx \sum_{i=1}^{NN} \vec{u}_i N_i \quad (0.35)$$

into equation (0.34) gives

$$\sum_{i=1}^{NN} \left(u_i \int_{\Omega} \frac{\partial N_i}{\partial x} d\Omega + v_i \int_{\Omega} \frac{\partial N_i}{\partial y} d\Omega \right) = 0, \quad (0.36)$$

where NN is the total number of velocity nodes in the discretized domain and N_i is the basis functions for velocity.

The summation in equation (0.36) is effectively a boundary node constraint equation because all internal nodes cancel in a finite element assembly. Applying Green's theorem in the following form,

$$\begin{aligned}\int_{\Omega} \frac{\partial N_i}{\partial x} d\Omega &= \int_{\Gamma} N_i \eta_x d\Gamma, \\ \int_{\Omega} \frac{\partial N_i}{\partial y} d\Omega &= \int_{\Gamma} N_i \eta_y d\Gamma,\end{aligned}\tag{0.37}$$

where $d\Gamma$ is an arc length of the domain boundary Γ . Thus, from equations (0.36) and (0.37),

$$\sum_{i=1}^{NN} \left(u_i \int_{\Gamma} N_i \eta_x d\Gamma + v_i \int_{\Gamma} N_i \eta_y d\Gamma \right) = 0.\tag{0.38}$$

With $N_i = 0$ on Γ for all internal nodes, the sum over NN is collapsed to a sum over NB , the number of boundary nodes.

The next step is to relate u and v to u_{η} and u_{τ} . Inserting equation (0.32) into equation (0.36) with the results of equation (0.38) yields,

$$\sum_{i=1}^{NB} \left[(\eta_{xi} u_{\eta i} - \eta_{yi} u_{\tau i}) \int_{\Omega} \frac{\partial N_i}{\partial x} d\Omega + (\eta_{xi} u_{\eta i} - \eta_{yi} u_{\tau i}) \int_{\Omega} \frac{\partial N_i}{\partial y} d\Omega \right] = 0,\tag{0.39}$$

which involves only those nodes of Γ and can be rearranged to

$$\sum_{i=1}^{NB} \left[\left(\eta_{xi} \int_{\Omega} \frac{\partial N_i}{\partial x} d\Omega + \eta_{yi} \int_{\Omega} \frac{\partial N_i}{\partial y} d\Omega \right) u_{\eta i} + \left(\eta_{xi} \int_{\Omega} \frac{\partial N_i}{\partial y} d\Omega - \eta_{yi} \int_{\Omega} \frac{\partial N_i}{\partial x} d\Omega \right) u_{\tau i} \right] = 0.\tag{0.40}$$

The mass consistent definition of the normal direction follows from equation (0.40) which, being a statement of global mass conservation, must be independent of the nodal tangential velocities. By operating on the tangential components of equation (0.40), it must then follow that

$$\eta_{xi} \int_{\Omega} \frac{\partial N_i}{\partial y} d\Omega = \eta_{yi} \int_{\Omega} \frac{\partial N_i}{\partial x} d\Omega.\tag{0.41}$$

When equation (0.41) is combined with the normalization requirement of a unit vector,

$$\eta_{xi}^2 + \eta_{yi}^2 = 1,\tag{0.42}$$

and substituted into equation (0.40), the components of the mass consistent outward pointing unit normal vector at boundary node i are then

$$\begin{aligned}\eta_{xi} &= \frac{1}{\eta_i} \int_{\Omega} \frac{\partial N_i}{\partial x} d\Omega, \\ \eta_{yi} &= \frac{1}{\eta_i} \int_{\Omega} \frac{\partial N_i}{\partial y} d\Omega,\end{aligned}\tag{0.43}$$

where η_i is the magnitude of $\vec{\eta}$ defined by

$$\eta_i = \left[\left(\int_{\Omega} \frac{\partial N_i}{\partial x} d\Omega \right)^2 + \left(\int_{\Omega} \frac{\partial N_i}{\partial y} d\Omega \right)^2 \right]^{1/2}.\tag{0.44}$$